



TESIS TE 142599

PENERAPAN SAGE (SPACE ALTERNATING GENERALIZED EXPECTATION-MAXIMIZATION) UNTUK ESTIMASI PARAMETER KANAL DUA ARAH (DOUBLE DIRECTIONAL) MENGGUNAKAN ANTENA ARRAY PLANAR

Musayyanah
2213203015

DOSEN PEMBIMBING
Dr. Ir. Puji Handayani, MT
Prof. Ir. Gamantyo Hendrantoro, M.Eng., Ph.D

PROGRAM MAGISTER
BIDANG KEAHLIAN TELEKOMUNIKASI MULTIMEDIA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2015



TESIS TE 142599

IMPLEMENTATION of SAGE (SPACE ALTERNATING GENERALIZED EXPECTATION-MAXIMIZATION) FOR ESTIMATE CHANNEL DOUBLE DIRECTIONAL PARAMETERS USING PLANAR ARRAY ANTENA

Musayyanah
2213203015

PROMOTORS

Dr. Ir. Puji Handayani, MT
Prof. Ir. Gamantyo Hendrantoro, M.Eng., Ph.D

**MAGISTER PROGRAM
DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY of INDUSTRIAL TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2015**

Tesis telah disusun untuk memenuhi salah satu syarat memperoleh gelar

Magister Teknik (MT)

Di

Institut Teknologi Sepuluh Nopember

oleh :

MUSAYYANAH

Nrp. 2213203015

Tanggal Ujian : 17 Juni 2015

Periode Wisuda : September 2015

Disetujui oleh:


1. Dr. Ir. Puji Handayani, MT
NIP : 196605101992032002

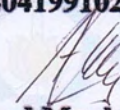
(Pembimbing 1)


2. Prof. Ir. Gamantyo Hendrantoro, M.Eng., Ph. D.
NIP : 197011111993031002

(Pembimbing 2)


3. Dr. Ir. Endroyono, DEA.
NIP : 196504041991021001

(Penguji 1)


4. Dr. Ir. Achmad Mauludiyanto, MT.
NIP : 196109031989031001

(Penguji 2)


5. Dr. Ir. Suwadi, MT.
NIP : 196808181993031002

(Penguji 3)



Direktur Program Pascasarjana,

Prof. Dr. Ir. Adi Soeprijanto, MT

NIP. 196404051990021001

PENERAPAN SAGE (SPACE ALTERNATING GENERALIZED EXPECTATION-MAXIMIZATION) UNTUK ESTIMASI PARAMETER KANAL DUA ARAH (DOUBLE DIRECTIONAL) MENGGUNAKAN ANTENA ARRAY PLANAR

Name : Musayyanah

NRP : 2213203015

Advisor : Dr. Ir. Puji Handayani, MT

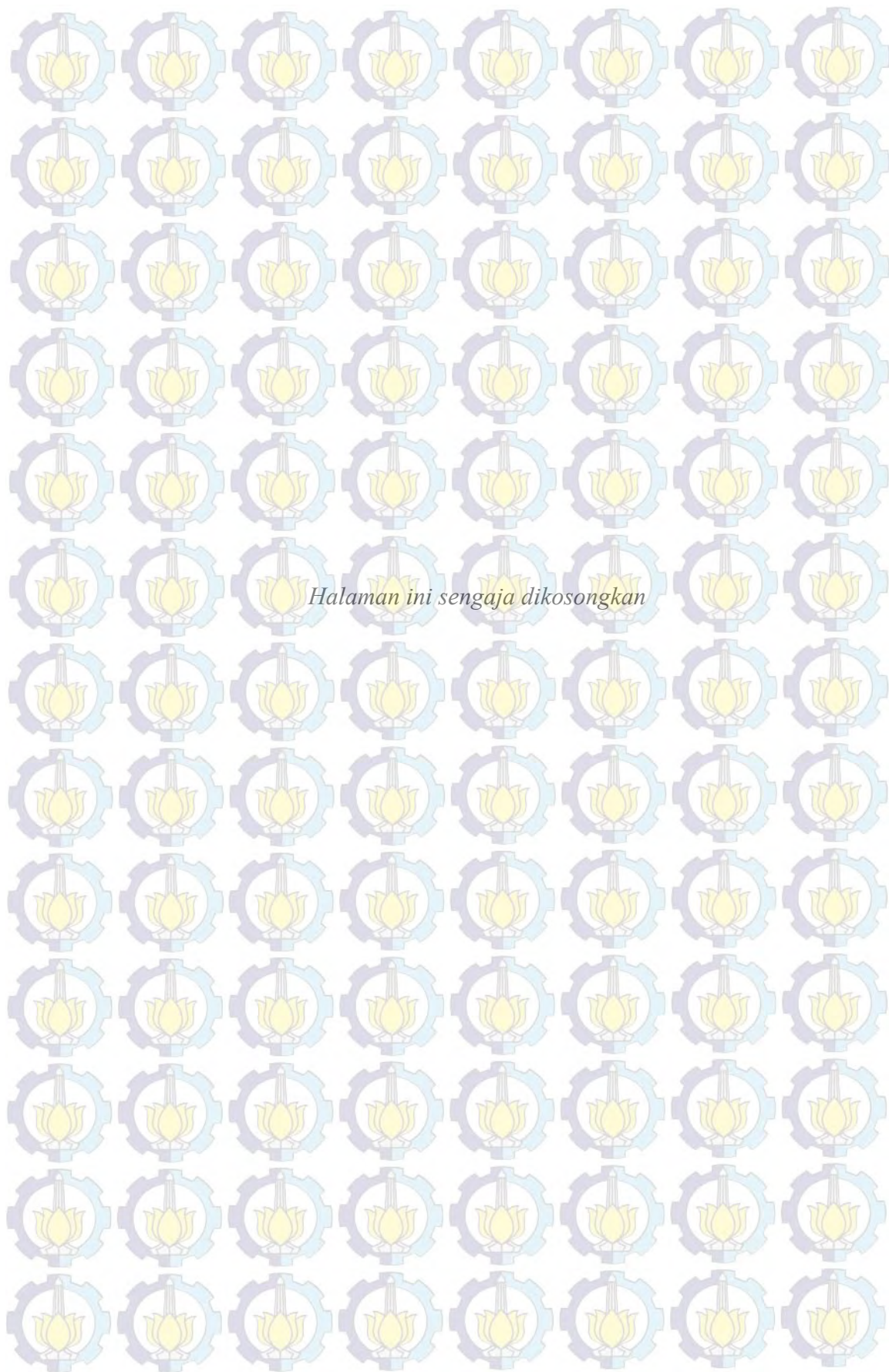
: Prof.Ir. Gamantyo Hendrantoro, M.Eng., Ph.D

ABSTRAK

Metode estimasi parameter *Space Alternating Generalized Expectation-Maximization* (SAGE) , mengestimasi parameter *subspace* yang terdiri dari *Directional Of Arrival* (DOA) dan *Directional of Departure* (DOD), dan parameter temporal yang terdiri dari Delay dan Amplitudo Kompleks. SAGE diterapkan dengan mengestimasi parameter sudut terlebih dahulu kemudian dilanjutkan dengan parameter temporal secara berurutan, hal ini dilakukan untuk pengukuran dengan perangkat resolusi delay yang rendah.

Hasil simulasi diterapkan pada antena array planar dan antena array linier, yang dinyatakan dengan metode RMSE. Hasil simulasi tersebut menunjukkan bahwa kinerja SAGE dalam mendeteksi parameter yang dimulai dari sudut lebih baik dibandingkan dengan kinerja SAGE yang mendeteksi parameter yang dimulai dari delay. Sedangkan hasil verifikasi SAGE dilakukan dengan pengukuran di Gedung Baru Hall PENS dengan menggunakan perangkat USRP dan antena array linier, dimana hasil estimasi delay didapatkan dari hasil crosscorelasi antara data yang diterima dan data yang dikirim yang selanjutnya terdapat 3 komponen sinyal yang terdeteksi hal ini disebabkan oleh resolusi perangkat yang rendah. Selain itu, SAGE mampu mengidentifikasi arah keberangkatan dan arah kedatangan dari 3 komponen sinyal tersebut. Arah keberangkatan dan arah kedatangan yang merupakan output dari SAGE, divalidasi ulang menggunakan *Ray Tracing 2 D* yang merupakan gambaran penajakan sinyal berdasarkan lokasi pengukuran. Hasil validasi antara output SAGE dan *Ray Tracing* mempunyai selisih eror sekitar 36° .

Kata Kunci : SAGE; *double directional*; *Directional of Arrival*; *Directional of Departure*; USRP



**IMPLEMENTATION of SAGE (SPACE ALTERNATING GENERALIZED
EXPECTATION-MAXIMIZATION) FOR ESTIMATE CHANNEL
DOUBLE DIRECTIONAL PARAMETERS USING PLANAR ARRAY
ANTENNA**

Name : Musayyanah
NRP : 2213203015
Advisor : Dr. Ir. Puji Handayani, MT
: Prof.Ir. Gamantyo Hendrantoro, M.Eng., Ph.D

ABSTRACT

Alternating Space parameter estimation method Generalized Expectation-Maximization (SAGE), estimating the parameters subspace consisting of Directional Of Arrival (DOA) and Directional of Departure (DOD), and temporal parameters consisting of Delay and Amplitude Complex. SAGE is applied to estimate the parameters of the corner first and then proceed with the temporal parameters sequentially, this is done to measure the delay that low-resolution devices.

The simulation results applied to a planar antenna array and linear array antenna, which is expressed by the method of RMSE. The simulation results show that the performance of SAGE detect parameters which starts from the point better than the performance of SAGE that detect parameters which starts from the delay. While the results of the verification is done by measuring SAGE in New Building Hall PENS using USRP devices and linear array antenna, in which the estimation results obtained from the crosscorrelasi delay between the received data and the subsequent data sent, there are 3 components of the detected signal caused by the resolution of this case Low devices. In addition, SAGE is able to identify the direction of departure and arrival directions of the three components of the signal. Direction of departure and arrival direction which is the output of SAGE, revalidated use Ray Tracing 2 D which is a picture based on the location of the measurement signal assessment. Validation results between SAGE and Ray Tracing output has an error margin of about 36 °.

Keywords : SAGE; *double directional; Directional of Arrival; Directional of Departure; USRP*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Assalamu'alaikum Wr. Wb

Dengan Nama Allah Yang Maha Penguasa dan Maha Penyayang, penulis mengucapkan Alhamdulillahirobbilalamin, atas bantuan dan kemudahan-NYA telah menyelesaikan penyusunan tesis yang berjudul :

PENERAPAN SAGE (SPACE ALTERNATING GENERALIZED EXPECTATION-MAXIMIZATION) UNTUK ESTIMASI PARAMETER KANAL DUA ARAH (DOUBLE DIRECTIONAL) MENGGUNAKAN ANTENA ARRAY PLANAR

Tesis ini digunakan sebagai salah satu syarat akademis untuk memperoleh gelar Magister Teknik (MT) pada Program Studi Teknik Elektro, Bidang Keahlian Telekomunikasi Multimedia, Institut Teknologi Sepuluh Nopember. Penyusun berharap semoga tesis ini dapat membawa manfaat untuk perkembangan penelitian selanjutnya. Selain itu, proses penyelesaian tesis ini tidak lepas dari banyak pihak yang telah memberikan saran dan masukan untuk terselesainya tesis ini. penyusun ingin menyampaikan terimakasih sedalam-dalamnya kepada pihak tersebut, di antaranya adalah :

1. Alm. Ayah dan Ibu yang selalu menjadi motivasi terbesar bagi penulis untuk terus mengejar ilmu sebanyak-banyaknya dan menjadi pribadi yang bermanfaat bagi orang lain.
2. Bu Puji Handayani dan Pak Gamantyo Hendrantoro atas bimbingan, kesabaran dan pendorong semangat dalam menyelesaikan tesis ini.
3. Bapak dan Ibu dosen yang terlibat dalam progres mingguan Lab Antena dan Propagasi B 306 atas masukan yang diberikan serta kritik yang membangun
4. Bapak Ahmad Zainuddin selaku dosen Politeknik Elektronika Negeri Surabaya (PENS) atas perizinan menggunakan fasilitas yang tersedia di PENS.
5. Bapak Endroyono, selaku dosen wali yang telah membimbing selama perkuliahan.

6. Anggota team Array (Martienda Erste Anggraeni, Mohammad Fadhli, Tania, dan Ranita) atas bantuan dan kerja sama serta motivasi yang telah diberikan.
7. Bapak dan Ibu dosen S2 dan S1 terima kasih atas ilmu yang diberikan selama perkuliahan di Intitut Teknologi Sepuluh Nopember
8. Rekan-rekan S2 dan S1 di Lab Antena dan Propagasi B-306, terima kasih atas kebaikan dan kerjasamanya dalam penelitian ini.

Penulis menyadari bahwa dalam penulisan tesis ini masih jauh dari sempurna, untuk itu demi perbaikan dan penyempurnaan tesis, maka kritik dan saran sangat diharapkan. Besar harapan penulis bahwa buku tesis ini dapat memberikan informasi dan manfaat bagi pembaca pada umumnya dan mahasiswa Jurusan Teknik Elektro pada khususnya.

Surabaya, Juli 2015

Penulis,
Musayyanah

DAFTAR ISI

| | |
|--|-----|
| HALAMAN JUDUL..... | i |
| PERNYATAAN KEASLIAN TESIS..... | ii |
| LEMBAR PENGESAHAN | iii |
| ABSTRAK | v |
| ABSTRACT..... | vii |
| KATA PENGANTAR..... | ix |
| DAFTAR ISI..... | xi |
| DAFTAR GAMBAR | xv |
| DAFTAR TABEL..... | xix |
| BAB 1 | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 3 |
| 1.3 Batasan Masalah..... | 3 |
| 1.4 Tujuan..... | 3 |
| 1.5 Manfaat Penelitian..... | 4 |
| BAB 2 | 6 |
| 2.1 Teori Penunjang | 6 |
| 2.1.1 Pseudonoise Code (PN Code) | 5 |
| 2.1.2 Modulasi Binary Phase Shift Keying..... | 7 |
| 2.1.3 Antena Array..... | 9 |
| 2.1.3.1 Antena Array Linier..... | 10 |
| 2.1.3.2 Antena Array Planar | 12 |
| 2.1.4 Model Lintasan..... | 13 |
| 2.1.5 Metode Estimasi Parameter Kanal | 15 |

| | | |
|---------|--|----|
| 2.1.6 | Algoritma SAGE | 15 |
| 2.1.6.1 | Expectation Steo (E-Step)..... | 15 |
| 2.1.6.2 | Maximization Step (M-Step) | 17 |
| 2.1.7 | Software Defined Radio (SDR)..... | 18 |
| 2.1.7.1 | USRP | 19 |
| 2.1.7.2 | Antena Dual Band | 21 |
| 2.18 | Verifikasi Algoritma SAGE | 22 |
| 2.19 | Proses Korelasi Untuk Mendapatkan Respon Impuls..... | 23 |
| 2.2 | Penelitian Sebelumnya | 25 |
| BAB 3 | | 27 |
| 3.1 | Asumsi Posisi Antena Planar/Linier..... | 28 |
| 3.2 | Pembangkitan PN Sequences | 30 |
| 3.3 | Simulasi Estimasi DOA, DOD, dan Amplitudo Kompleks | 31 |
| 3.4 | Hasil Simulasi..... | 36 |
| 3.5 | Pengukuran Respon Kanal..... | 37 |
| 3.6 | Setup Pengukuran | 38 |
| 3.7 | Pemrosesan Data..... | 39 |
| BAB 4 | | 46 |
| 4.1 | Akurasi Estimasi Algoritma SAGE sebagai Fungsi SNR | 44 |
| 4.1.1 | Perubahan SNR pada Antena Array Planar | 46 |
| 4.1.2 | Perubahan SNR pada Antena Array Linier | 49 |
| 4.2 | Akurasi Estimasi Algoritma SAGE sebagai Fungsi Jumlah Multipath Sinyal.. ... | 52 |
| 4.2.1 | Perubahan Jumlah Multipath Sinyal Pada Antena Array Planar...52 | |
| 4.2.2 | Perubahan Jumlah Multipath sinyal Pada Antena Array Linier ... | 56 |
| 4.3 | Akurasi Estimasi Algoritma SAGE sbg Beda Sudut Elevasi..... | 59 |
| 4.3.1 | Perubahan Sudut Elevasi Pada Antena Array Planar..... | 59 |
| 4.3.2 | Perubahan Sudut Elevasi Pada Antena Array Linier | 61 |

| | | |
|-----------------------------|--|------------|
| 4.4 | Kelebihan Kinerja SAGE dalam Mengestimasi Parameter Sudut Dulu | 64 |
| 4.5 | Validasi Algoritma SAGE berdasarkan Pengukuran di Hall Gedung Baru Politeknik Elektronika Negeri Surabaya | 66 |
| BAB 5 | | 70 |
| 5.1 | Kesimpulan..... | 70 |
| 5.2 | Saran | 72 |
| DAFTAR PUSTAKA | | 75 |
| LAMPIRAN..... | | 77 |
| RIWAYAT PENULIS..... | | 114 |

Halaman ini sengaja dikosongkan

DAFTAR TABEL

| | |
|---|----|
| Tabel 2.1 Feedback M-Sequence | 6 |
| Tabel 2.2 Spesifikasi Daughterboard | 19 |
| Tabel 3.1 Setup Pengukuran | 39 |
| Tabel 3.2 Urutan Penyimpanan data Pengukuran | 40 |
| Tabel 4.1 RMSE Sudut terhadap perubahan nilai SNR (Antena Planar) | 46 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 M-Sequence..... | 5 |
| Gambar 2.2 Ilustrasi Gold Code..... | 6 |
| Gambar 2.3 Konstelasi BPSK..... | 7 |
| Gambar 2.4 Modulator BPSK..... | 8 |
| Gambar 2.5 Demodulator BPSK..... | 8 |
| Gambar 2.6 Konfigurasi Array Linier..... | 9 |
| Gambar 2.7 Array Linier Colateral | 10 |
| Gambar 2.8 Pola Radiasi Array Linier Colateral..... | 11 |
| Gambar 2.9 Geometry dari Array Planar | 11 |
| Gambar 2.10 Lintasan Propagasi Sinyal dari TX ke RX | 13 |
| Gambar 2.11 SFG Algoritma SAGE..... | 18 |
| Gambar 2.12 Daughterboard SBX..... | 19 |
| Gambar 2.13 Blok Diagram USRP N2922 | 20 |
| Gambar 2.14 Konfigurasi USRP N210 dan N2922 menggunakan kabel MIMO.. | 21 |
| Gambar 2.15 Antena Monopole Dualband | 21 |
| Gambar 2.16 Metode Refleksi pada Ray Tracing..... | 22 |
| Gambar 3.1 Diagram Alir Penelitian | 27 |
| Gambar 3.2 Posisi Elemen Antena Array Planar | 28 |
| Gambar 3.3 Posisi Elemen Antena Array Linier | 29 |
| Gambar 3.4 Pola Radiasi Antena Array Linier | 29 |
| Gambar 3.5 Flowchart Simulasi SAGE | 33 |
| Gambar 3.6 Blok Diagram Pengukuran..... | 38 |
| Gambar 3.7 Proses Pengolahan Data | 40 |
| Gambar 3.8 Respom Impuls (H_{11} ; H_{12} ; H_{21} ; H_{22}) | 39 |
| Gambar 3.9 Grafik Hasil Pengecekan Konsistensi..... | 43 |
| Gambar 3.10 Respon Impuls MIMO..... | 44 |
| Gambar 4.1 Kinerja SAGE terhadap Variasi Nilai SNR Untuk Estimasi Parameter Delay pada Antena Planar..... | 46 |

| | |
|--|----|
| Gambar 4.2 Kinerja SAGE terhadap Variasi Nilai SNR Untuk Estimasi Parameter Sudut pada Antena Array Planar..... | 47 |
| Gambar 4.3 Kinerja SAGE terhadap Variasi Nilai SNR Untuk Estimasi Amplitudo Kompleks Sudut pada Antena Array Planar | 48 |
| Gambar 4.4 Kinerja SAGE terhadap Variasi Nilai SNR untuk Estimasi Parameter Delay pada Antena Array Linier..... | 49 |
| Gambar 4.5 Kinerja SAGE terhadap Variasi Nilai SNR untuk Estimasi Parameter Sudut pada Antena Array Linier..... | 50 |
| Gambar 4.6 Kinerja SAGE terhadap Variasi nilai SNR untuk Estimasi Parameter Amplitudo Kompleks pada Antena Array Linier..... | 51 |
| Gambar 4.7 Kinerja SAGE terhadap Variasi Jumlah Sinyal untuk Estimasi Delay pada Antena Array Planar..... | 53 |
| Gambar 4.8 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut Elevasi pada Antena Array Planar | 54 |
| Gambar 4.9 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut Azimuth pada Antena Array Planar..... | 54 |
| Gambar 4.10 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Amplitudo Kompleks pada Antena Array Planar..... | 55 |
| Gambar 4.11 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Delay pada Antena Array Linier..... | 56 |
| Gambar 4.12 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut (DOA dan DOD) pada Antena Array Linier | 57 |
| Gambar 4.13 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Amplitudo Kompleks pada Antena Array Linier..... | 58 |
| Gambar 4.14 Kinerja SAGE Terhadap Beda Sudut DOA pada Antena Array Planar. | 59 |
| Gambar 4.15 Kinerja SAGE Terhadap Beda Sudut DOD pada Antena Array Planar..... | 60 |
| Gambar 4.16 Kinerja SAGE Terhadap Perubahan Sudut Elevasi DOA pada Antena Array Linier..... | 61 |
| Gambar 4.17 Kinerja SAGE terhadap Perubahan Sudut Elevasi DOD pada Antena Array Linier..... | 61 |

| | |
|--|----|
| Gambar 4.18 Kinerja SAGE terhadap Beda Sudut DOA pada Antena Array Linier | 62 |
| Gambar 4.19 Kinerja SAGE terhadap Beda Sudut DOD pada Antena Array Linier | 63 |
| Gambar 4.20 Kinerja SAGE Mengestimasi Sudut Terhadap Variasi Komponen dengan Delay yang sama Pada Antena Array Planar..... | 64 |
| Gambar 4.21 Kinerja SAGE Mengestimasi Sudut Terhadap Variasi Komponen dengan Delay yang sama Pada Antena Array Linier..... | 65 |
| Gambar 4.22 Ray Tracing dari Lintasan yang Terdeteksi oleh SAGE..... | 66 |
| Gambar 4.23 Respon Impuls Hasil Pengukuran..... | 67 |

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan komunikasi bergerak membutuhkan peningkatan kapasitas kanal dan kecepatan layanan transmisi data yang tinggi sehingga membutuhkan peningkatan pengembangan sistem transmisi multi antena. Pengembangan tersebut membutuhkan informasi yang lengkap mengenai model realistis dari karakteristik kanal untuk mendapatkan parameter fisik dari perambatan gelombang radio. Parameter fisik tersebut terdiri dari arah keberangkatan (DOD) dan arah kedatangan (DOA), waktu delay, dan amplitudo kompleks yang perlu diestimasi untuk mendapatkan karakter model yang realistis kanal yang diinginkan. Selain itu arah keberangkatan dan arah kedatangan merupakan parameter spasial yang terdiri dari komponen elevasi dan azimuth, sedangkan waktu delay dan amplitudo kompleks merupakan parameter temporal.

Parameter fisik tersebut didapatkan dengan menggunakan metode atau algoritma tertentu. Pada [1], Fleury melaporkan bahwa metode tersebut digolongkan menjadi dua yaitu *Parametric Subspace-Based Estimation* (PSBE) dan *Deterministic Parametric Estimation* (DPE). Salah satu contoh dari metode PSBE adalah *Multiple Signal Classification* (MUSIC) dan *Estimation Signal Parameter Via Rotation Invariance Techniques* (ESPRIT).

Pada [2], Syahroni melaporkan bahwa MUSIC memiliki keterbatasan dalam mengestimasi parameter-parameter sinyal yang jumlahnya dibatasi oleh jumlah antena yang digunakan. Sedangkan metode ESPRIT memiliki keterbatasan terhadap bentuk geometri antena yang digunakan dan jumlah sinyal [3].

Contoh dari metode DPE, adalah algoritma Expectation Maximization, EM yang memiliki tingkat konvergensi yang rendah dengan inisialisasi yang tinggi sehingga membuat algoritma ini membutuhkan waktu yang lama untuk proses komputasinya, serta komputasi akurasi dari algoritma ini rendah untuk mendapatkan parameter fisik kanal tersebut. Kelemahan dari EM diperbaiki oleh algoritma Space Alternating Generalized EM, SAGE yang memiliki kecepatan

konvergensi yang lebih tinggi dan lebih akurat. Selain itu, SAGE dapat diterapkan di geometri sebarang dari antenna array.

SAGE dianggap memiliki kemampuan memisahkan sinyal dalam lintasan jamak pada kanal dua arah (*double directional channel*). Pada [4] telah menerapkan SAGE menggunakan antenna biconical $\lambda/2$ volume kubus sintetik di sisi pemancar dan penerima dan pada [5] menggunakan antenna array $\lambda/2$ 8 elemen Uniform Linier Array (ULA) di sisi pemancar dan 4×4 *Uniform Square Array* (USA) di sisi penerima. Sedangkan pada [6] menggunakan antenna array *Uniform Linear Patch Array* (ULPA) pada sisi tx, dan di sisi rx menggunakan 16 elemen *Uniform Circular Array* (UCA).

Banyak penelitian yang menggunakan algoritma SAGE untuk mendapatkan parameter fisik kanal tersebut, yang diterapkan pada kanal sintetik dan real. Pada kanal sintetik dilakukan simulasi pengujian kinerja SAGE sebagai algoritma yang memiliki resolusi yang tinggi untuk mendapatkan parameter temporal dan parameter spasial secara berurutan, yang kemudian diterapkan pada kanal real untuk mendapatkan parameter-parameter tersebut.

Implementasi SAGE pada kanal real tergantung dari resolusi peralatan yang digunakan. Pada [4] implementasi SAGE dilakukan dengan menggunakan *Virtual Network Analyzer* (VNA) dengan resolusi temporal (resolusi delay) yang sesuai. Implementasi tersebut dilakukan pada domain frekuensi yang terbatas terhadap lingkungan statis dan membutuhkan waktu yang lama karena menggunakan array sintetik. Keterbatasan tersebut dapat diatasi dengan menggunakan pengukuran kanal real time yang menggunakan perangkat *Universal Software Radio Peripheral* (USRP).

USRP merupakan perangkat yang bisa digunakan untuk pengukuran kanal dengan mengirimkan deretan pn sequence. Keterbatasan yang dimiliki oleh USRP adalah resolusi temporal dari perangkat tersebut cukup besar apabila digunakan untuk karakteristik kanal, sehingga parameter delay tidak terdeteksi dengan baik. Langkah awal untuk mengatasi hal tersebut adalah membuat simulasi SAGE untuk mendapatkan parameter fisik kanal, dimana pendeteksian parameter tersebut dimulai dari parameter spasial dan dilanjutkan dengan parameter temporal.

1.2 Rumusan Masalah

Permasalahan yang harus diselesaikan pada penelitian ini adalah :

1. Bagaimana tingkat akurasi SAGE apabila urutan parameter fisik kanal diubah dimulai dari parameter spasial kanal dilanjutkan dengan parameter temporal.
2. Bagaimana akurasi algoritma SAGE untuk estimasi parameter fisik dari kanal terhadap pengaruh SNR, jumlah lintasan sinyal, dan beda/ selisih DOA dan DOD yang menggunakan antena array planar dan antena linier
3. Bagaimana perbandingan kinerja SAGE pada kanal sintetis (simulasi) dan kanal real (pengukuran)?

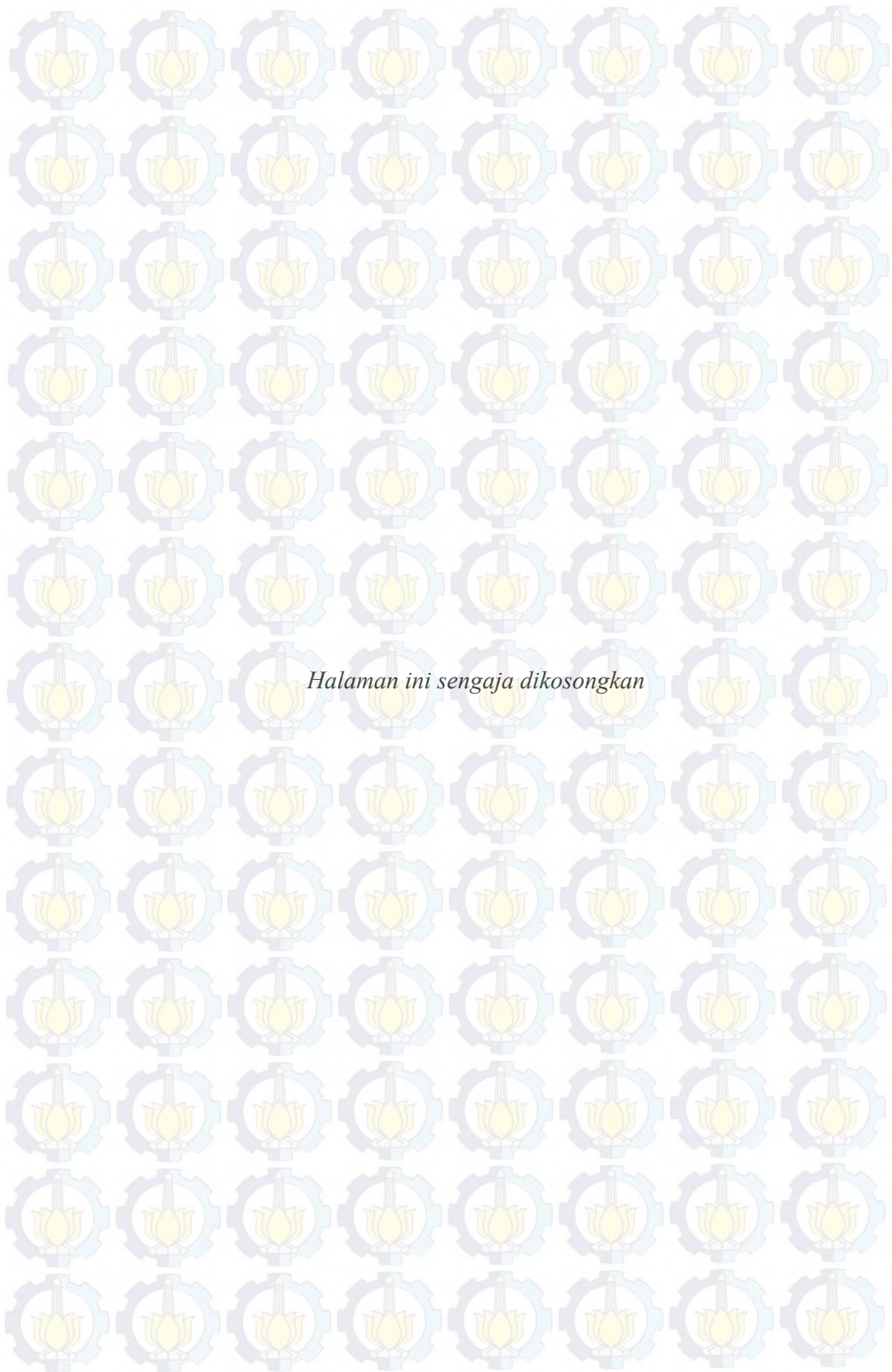
1.3 Batasan Masalah

1. Modulasi yang digunakan adalah *Binary Phase Shift Keying* (BPSK)
2. Noise yang dibangkitkan berdistribusi *Additive White Gaussian Noise* (AWGN) dengan *mean* 0 dan *variance* 1.
3. Simulasi kinerja SAGE diterapkan pada antena 4x4 array planar dan antena 2x2 linier di sisi pemancar dan penerima
4. Parameter fisik kanal yang diestimasi adalah arah kedatangan sinyal (DOA), arah keberangkatan sinyal (DOD), waktu delay, dan amplitudo kompleks
5. Validasi simulasi dilakukan dengan konsep pengukuran *Time Division Multiplexing* (TDMA) yang menggunakan *Universal Software Radio Peripheral* (USRP) N2922 pada frekuensi 2.475 GHz dengan antena linier *omnidirectional* 2x2 di sisi pemancar dan penerima

1.4 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

Melakukan estimasi parameter fisik kanal yang terdiri dari arah kedatangan sinyal (DOA), arah keberangkatan sinyal (DOD), waktu delay, dan amplitudo kompleks secara berurutan, yang diterapkan pada antenna array planar dan array antenna linier.



BAB 2

TINJAUAN PUSTAKA

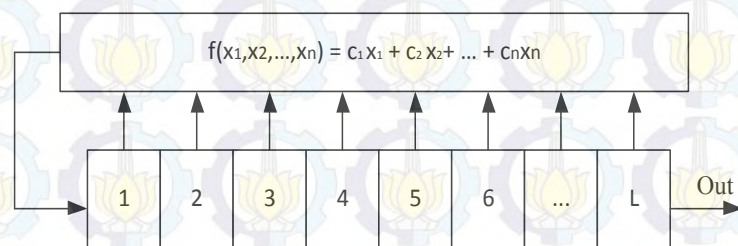
2.1 Teori Penunjang

Pada bab ini diuraikan teori penunjang yang terkait dengan penelitian di antaranya adalah PN- Sequences, Modulasi *Binary Phase Shift Keying* (BPSK), Antena Array, Algoritma SAGE, dan Ray Tracing.

2.1.1 Pseudonoise Code (PN Code)

Pseudo-Noise yang merupakan biner *suquence* dengan autokorelasi yang mirip dalam satu periode, yang mempunyai korelasi yang sangat rendah. PN sequence tidak benar-benar random, tapi merupakan sinyal periodik yang diketahui penerima maupun pengirim [7]. PN code dapat dibagi menjadi beberapa jenis berikut ini.

- *Maximum Length Sequence* (M-Sequence) yang dibangkitkan dari sebuah *Linier Feedback Shift Register*. Blok diagram dari M-sequence ditunjukkan pada gambar 2.1 berikut :



Gambar 2.1. M-Sequence [7]

Blok diagram *M-Sequence* merupakan *shift register* yang sederhana menerima sinyal *feedback* yang dikembalikan ke pemancar dengan cara di-XOR-kan. Panjang code *M-sequence* dirumuskan pada persamaan (2.1) sebagai berikut :

$$P = 2^L - 1 \quad (2.1)$$

dimana

L : jumlah *shift register* yang digunakan.

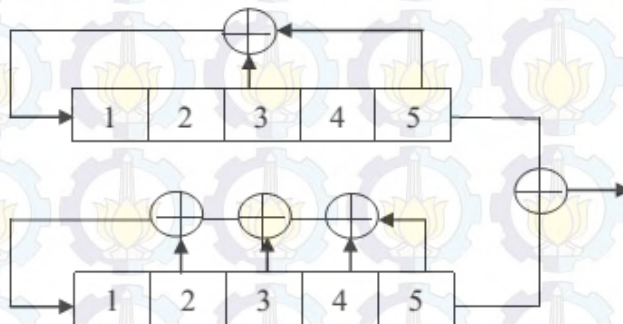
Berikut ini adalah tabel feedback m-sequence yang dihasilkan

Tabel 2.1. *Feedback M-Sequence*

| L | $N = 2^L - 1$ | Feedback Tap | M-sequence |
|---|---------------|--|------------|
| 2 | 3 | [2,1] | 2 |
| 3 | 7 | [3,1] | 2 |
| 4 | 15 | [4,1] | 2 |
| 5 | 31 | [5,3] [5,4,3,2] [5,4,2,1] | 6 |
| 6 | 63 | [6,1] [6,5,2,1] [6,5,3,2] | 6 |
| 7 | 127 | [7,1] [7,3] [7,3,2,1] [7,4,3,2] [7,6,4,2] [7,6,3,1] [7,6,5,2] [7,6,5,4,2] [7,5,4,3,2,1] | 18 |
| 8 | 255 | [8,4,3,2] [8,6,5,3] [8,6,5,2] [8,5,3,1] [8,6,5,1] [8,7,6,1] [8,7,6,5,2,1] [8,6,4,3,2,1] | 16 |
| 9 | 511 | [9,4] [9,6,4,3] [9,8,5,4] [9,8,4,1] [9,5,3,2] [9,8,6,5] [9,8,7,6,3,1] [9,8,7,6,5,3] | 48 |

- **Code Gold**

Code Gold adalah salah satu turunan dari M-sequence, yang disusun oleh 2 buah M-sequence yang masing-masing outputnya ditambahkan (*adder modulo 2*). Kedua output dari M-Sequence ditambahkan (XOR) secara *chip per chip* menggunakan pulsa-pulsa *clock* yang sinkron. Kedua M-Sequence mempunyai panjang yang sama. Panjang maksimal dari Gold Code adalah $2^L + 1$ (M-sequence itu sendiri) dan $2^L - 1$ (dari kombinasi yang dihasilkan dari pergeseran posisi chip), dengan L adalah *jumlah shift register*. Ilustrasi dari gold code ditunjukkan pada Gambar 2.2



Gambar 2.2 Ilustrasi Gold Code

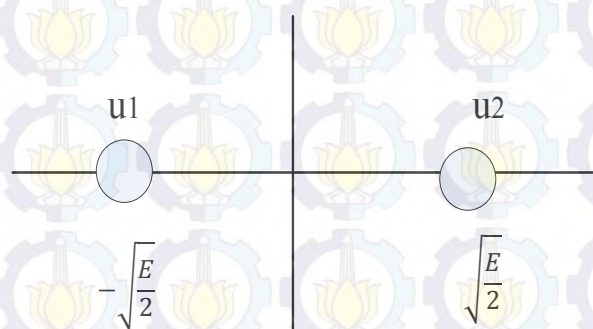
Apabila dipilih M-sequence khusus atau yang disebut *preferd pair*, maka *Gold Code* yang dibangkitkan mempunyai 3 nilai *cross correlation*, yaitu : $-1, -2^{[(L+2)/2]}, 2^{[(L+2)/2]} - 2$.

- **Code Kasami**

Kode Kasami merupakan penurunan dari *m-sequences* yang dapat menghemat sejumlah register (panjang code kasami lebih pendek) untuk nilai *threshold* yang sama dengan kode gold. Kasami code dibagi menjadi 2 yaitu *small set* dan *large set* dengan panjang kode maksimal keduanya adalah $2^L - 1$. Small set mempunyai 3 nilai *cross correlation*, yaitu $-1, -(2^{\frac{L}{2}} + 1), (2^{L/2} + 1) - 2$, sedangkan *large set* mempunyai 5 nilai *cross correlation* sebagai berikut : $-1, -(2^{L/2} + 1), (2^{L/2} + 1) - 2, -(2^{[(L+2)/2]} + 1), (2^{[(L+2)/2]} + 1) - 2$.

1.1.2 Modulasi *Binary Phase Shift Keying*

Modulasi BPSK merupakan salah satu jenis modulasi digital yang mengirimkan bit + 1 merepresentasikan bit 1, dan bit -1 merepresentasikan bit 0. BPSK mempunyai dimensi satu yang ditunjukkan dengan konstelasi dari BPSK pada Gambar 2.3 berikut :



Gambar 2.3 Konstelasi BPSK

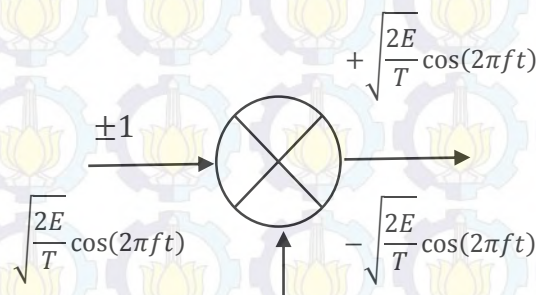
Apabila input yang diberikan +1, maka output dari modulasi BPSK mengalami pergeseran 0° . Jika input yang diberikan -1, maka output dari modulasi BPSK

mengalami pergeseran 180° . Output dari modulasi BPSK dapat dirumuskan pada persamaan (2.2) dan (2.3) sebagai berikut:

$$u_1 = \sqrt{\frac{2E}{T}} \sin(\omega t) \quad (2.2)$$

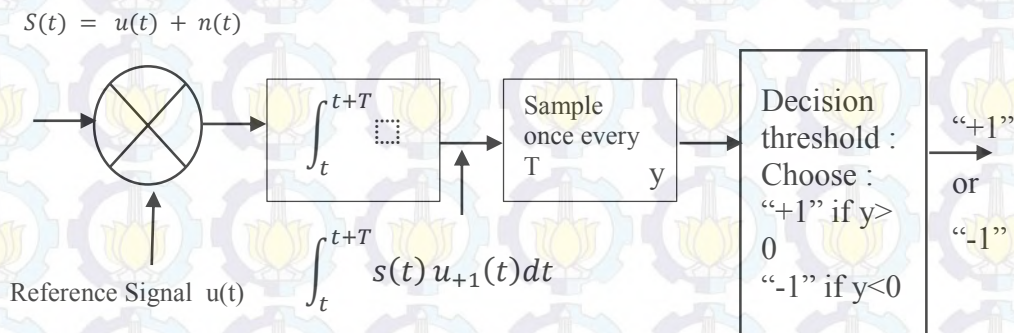
$$u_2 = \sqrt{\frac{2E}{T}} \sin(\omega t + 180^\circ) = -\sqrt{\frac{2E}{T}} \cos(\omega t) \quad (2.3)$$

Pemancar dan Penerima dari modulasi BPSK ditunjukkan pada Gambar 2.4 dan Gambar 2.5



Gambar 2.4. Modulator BPSK

Pada Gambar 2.4 menunjukkan modulator dari BPSK dengan input +1 dan -1, sedangkan Gambar 2.5 menunjukkan demodulator dari BPSK dengan mengasumsikan deteksi yang digunakan adalah *Maximum Likelihood Detector* (MLD) dengan probabilitas pengiriman +1 sama dengan probabilitas penerimaan -1. Pada Gambar 2.5 menunjukkan $s(t)$ adalah sinyal $u(t)$ yang diterima setelah ditambahkan noise $n(t)$ yang kemudian didemodulasi oleh demodulator BPSK untuk mendapatkan bit +1 atau -1.



Gambar 2.5 Demodulator [13]

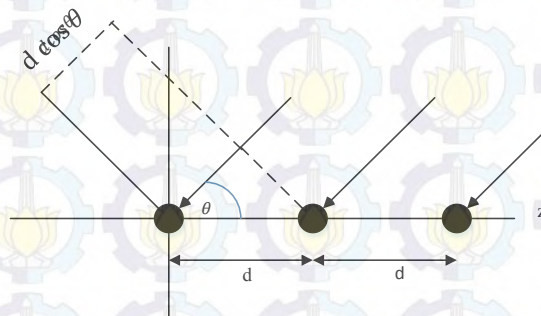
1.1.3 Antena Array

Antena array terdiri dari beberapa elemen yang diatur dalam jarak tertentu dan dihubungkan satu sama lain untuk membentuk suatu pola radiasi yang diinginkan. Pola radiasi setiap elemen antena dapat dikendalikan dengan menggabungkan beberapa elemen antena yang memungkinkan perubahan pola radiasi secara signifikan. Tujuan pembentukan antena array adalah meningkatkan gain dan daya radiasi, selain itu pembentukan antena array harus memperhatikan faktor terpenting seperti bentuk konfigurasi geometri dari antena array, jarak antar elemen, amplitudo dan fase dari masing-masing elemen antena dan pola radiasi dari individu elemen antena.

Analisis array mulai menggunakan elemen yang meradiasikan ke segala arah yang disebut sebagai sumber titik isotropis. Pemancar dan Penerima elemen isotropis mengarah ke segala arah dalam tiga dimensi [8]. Pola dari elemen array isotropis disebut sebagai *array factor*, yang menunjukkan perkalian array dengan dengan pola elemen untuk mendapatkan pola radiasi penuh.

1.1.3.1 Antena Array Linier

Konfigurasi dari antena array yang paling dasar adalah *Unifrom Linier Array* (ULA) yang disusun dengan jarak tertentu yang membentuk garis lurus yang ditunjukkan pada Gambar 2.6. Kita asumsikan bahwa *point source* adalah antenna isotropis yang mempunyai pola radiasi yang sama, dengan d adalah jarak antara elemen, θ adalah sudut antara sumbu z dengan sinyal yang datang dari medan jauh. Maka $d \cos \theta$ menunjukkan sinyal yang datang yang diterima oleh sebuah antena elemen.



Gamabr 2.6 Konfigurasi Array Linier

Jika eksitasi arus untuk n element adalah I_n , array factor untuk array antenna dengan N element didapatkan dari total dari I_n untuk setiap elemen antenna, array factor dirumuskan sebagai berikut,

$$AF = I_0 + I_1 e^{j\beta d \cos \theta} + I_2 e^{j\beta d \cos \theta} + \dots I_n e^{j\beta d \cos \theta} \quad (2.4)$$

Dimana $\beta = \frac{2\pi}{\lambda}$

Apabila terdapat linier phase antara elemen, maka I_n dirumuskan

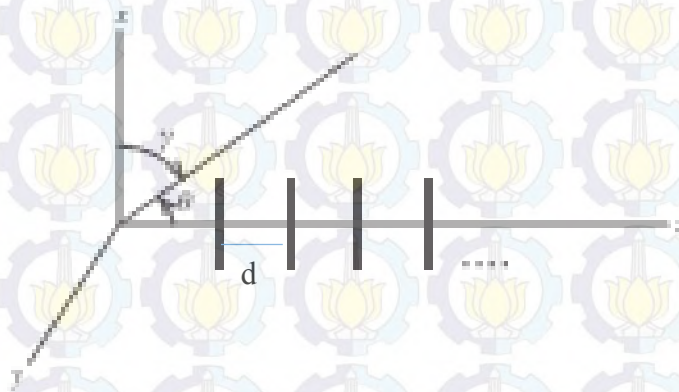
$$I_n = A_n e^{jn\alpha} \quad (2.5)$$

Dimana α adalah phase dari elemen $n+1$, sehingga array factor menjadi

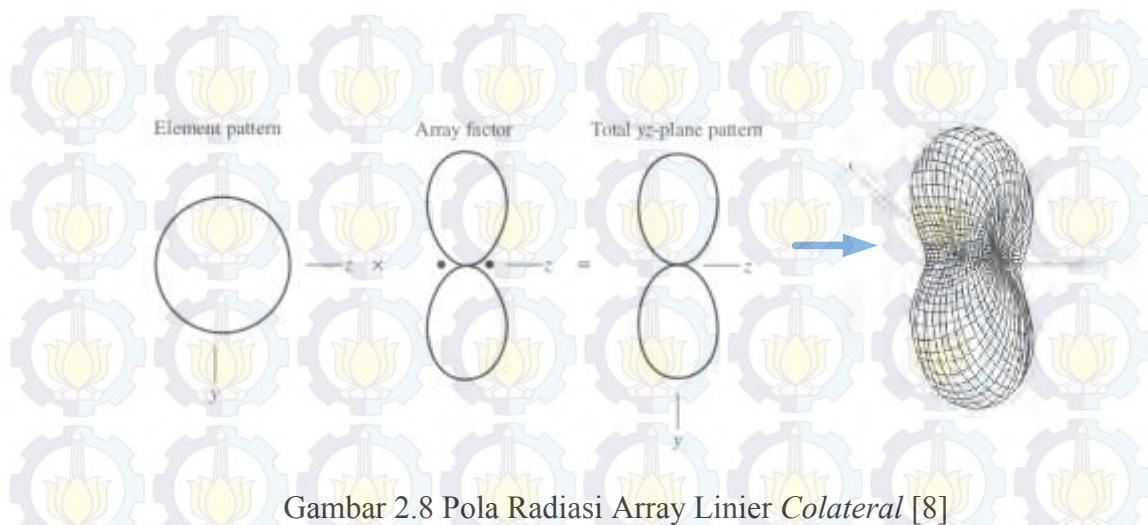
$$AF = \sum_{n=0}^{N-1} A_n e^{jn(\beta d \cos \theta + \alpha)} \quad (2.6)$$

Maka dari persamaan (2.6), array factor tergantung dari arus dan eksitasi phase dari setiap elemen antenna.

Beberapa elemen yang terletak sepanjang garis yang dialiri arus sepanjang garis tersebut disebut sebagai *colliner* dan *collateral*. Pada array *colliner* array elementnya garis yang disusun segaris (linier) dengan distribusi arus searah dengan garis, sedangkan array *collateral* merupakan kebalikan dari *colliner*. Jarak antara elemen yang terletak pada sumbu z , yang dipisahkan sebesar d jarak dengan arus yang dijumlahkan searah sumbu z . Array Linier *Collateral* beserta pola radiasinya ditunjukkan pada Gambar 2.7 dan 2.8



Gambar 2.7 Array Linier Colateral[8]

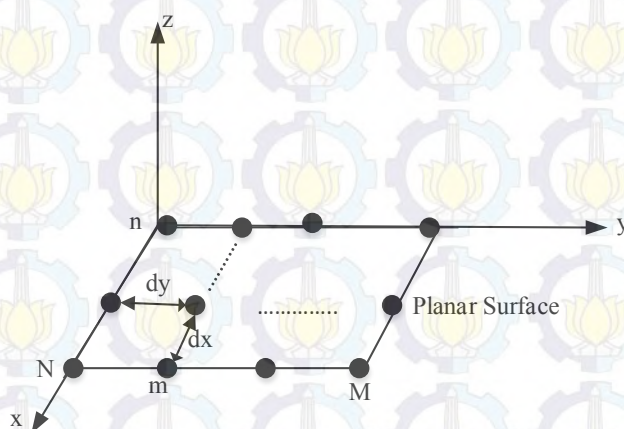


Gambar 2.8 Pola Radiasi Array Linier *Colateral* [8]

Pada gambar 2.8 menunjukkan salah satu contoh pola radiasi dari array linier colateral dengan menggunakan antenna dipole. Susunan array dengan konfigurasi di atas menunjukkan bahwa pola radiasi total juga dipengaruhi oleh pola elemen antenna, dimana elemen antenna tersebut juga dipengaruhi oleh posisi atau arah dari elemen antenna.

1.1.3.2 Antena Array Planar

Elemen dari Array Planar (Array Bidang) diletakkan dalam suatu bidang. Jenis konfigurasi ini dapat membentuk suatu bidang lingkaran atau bidang persegi dengan elemen terletak di pusat lingkaran dan di pusat bidang kotak. Selain itu terdapat konfigurasi elemen yang terletak pada permukaan yang tidak merata yang disebut sebagai array gabungan atau bidang volume misalnya kubus, prisma, dan *octahedron*.



Gambar 2.9 Geometry dari Array Planar

Pada Gambar 2.9 terlihat bahwa jarak antar elemen pada sumbu x dan jarak antar elemen pada sumbu y diasumsikan sama ($dx = dy$) yang merupakan bentuk geometri dari array planar atau array multidimensional yang membutuhkan *beam*, gain yang tinggi atau *scanning main beam* ke segala arah, dimana penggunaan array multidimensional dianggap menguntungkan. Array Multidimensional dibagi menjadi 3 kategori yaitu : geometri dengan elemen yang terletak pada pusat bidang (lingkaran, rectangular atau kotak) ,terletak pada keliling bidang yang biasanya berbentuk lingkaran, dan geometri grid (*lattice*) yang terdapat pengaturan jarak pusat elemen dengan elemen yang lain. dimana :

$$r'_{mn} = x'_{mn}\hat{x} + y'_{mn}\hat{y} + z'_{mn}\hat{z} \quad (2.7)$$

$$f(\theta, \phi) = \sum_{n=1}^N \sum_{m=1}^M I_{mn} e^{j\alpha_{mn}} e^{j\xi_{mn}} \quad (2.8)$$

$$\xi_{mn} = \frac{2\pi}{\lambda} [x'_{mn} \sin \theta \cos \phi + y'_{mn} \sin \theta \sin \phi + z'_{mn} \cos \phi]$$

$$\alpha_{mn} = -\frac{2\pi}{\lambda} [x'_{mn} \sin \theta_0 \cos \phi_0 + y'_{mn} \sin \theta_0 \sin \phi_0 + z'_{mn} \cos \phi_0]$$

$$\theta_0, \phi_0 = \text{arah main beam}$$

$$\lambda = \text{panjang gelombang}$$

$$I_{mn} = \text{amplitude arus eksitasi elemen ke mn}$$

Persamaan (2.7) menunjukkan bahwa *array factor* dipengaruhi oleh jumlah elemen, jarak elemen, amplitude eksitasi, dan fasa relative pada setiap elemennya. Dalam mengestimasi parameter kanal, khususnya seperti DOA dan DOD dapat dirumuskan sebagai arah datang ataupun arah berangkat dari suatu sinyal yang disebut dengan *steering vector* dirumuskan pada persamaan (2.9)

$$c(\theta, \phi) \triangleq e^{j\frac{2\pi}{\lambda}(e(\theta_L, \phi_L), r)} \quad (2.9)$$

dengan $\langle \dots, \dots \rangle$ menyatakan *inner product* dari dua vektor (unit vector dan vector posisi dari elemen antenna dalam bidang array), persamaan 2.9 dapat dipaparkan seperti persamaan (2.10) dan (2.11)

$$\mathbf{c}_1(\boldsymbol{\Omega}_1) = \left[f_{1,1}(\boldsymbol{\Omega}_1) e^{j\frac{2\pi}{\lambda}(\boldsymbol{\Omega}_1 \cdot \mathbf{r}_{1,1})}, \dots, f_{1,M}(\boldsymbol{\Omega}_1) e^{j\frac{2\pi}{\lambda}(\boldsymbol{\Omega}_1 \cdot \mathbf{r}_{1,M})} \right]^T \quad (2.10)$$

$$\mathbf{c}_2(\boldsymbol{\Omega}_2) = \left[f_{2,1}(\boldsymbol{\Omega}_2) e^{j\frac{2\pi}{\lambda}(\boldsymbol{\Omega}_2 \cdot \mathbf{r}_{2,1})}, \dots, f_{2,N}(\boldsymbol{\Omega}_2) e^{j\frac{2\pi}{\lambda}(\boldsymbol{\Omega}_2 \cdot \mathbf{r}_{2,N})} \right]^T \quad (2.11)$$

dimana :

$\mathbf{c}_1(\boldsymbol{\Omega}_1)$ = *steering vector* dari array planar di sisi pemancar pada M elemen

$\mathbf{c}_2(\boldsymbol{\Omega}_2)$ = *steering vector* dari array planar di sisi penerima pada N elemen

$f_{n,N}, f_{m,M}$ = pola medan antenna elemen di sisi pemancar dan penerima

λ = panjang gelombang

$\mathbf{r}_{n,N}$ = vector posisi elemen antenna penerima

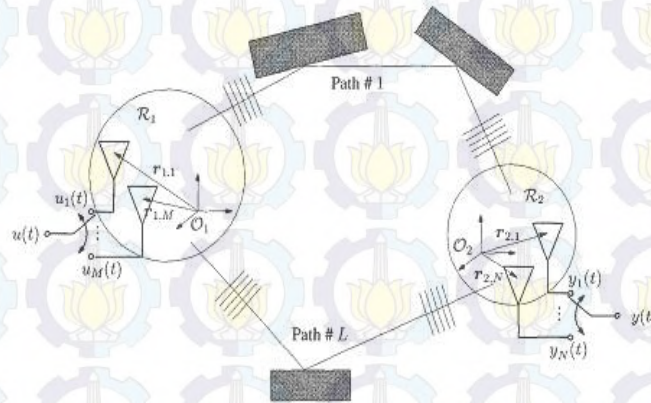
$\mathbf{r}_{m,M}$ = vector posisi elemen antenna pemancar

$\boldsymbol{\Omega}_n$ = unit vector dari koordinat bola $[\sin \theta_n \cos \phi_n, \sin \theta_n \sin \phi_n, \cos \theta_n]^T$,

dengan $\theta_n: 0^\circ - 180^\circ$ dan $\phi_n: 0 - 360^\circ$.

1.1.4 Model Lintasan

Propagasi sinyal dari pemancar ke penerima yang melewati beberapa lintasan *multipath* yang berbeda ($\ell = 1, 2, \dots$) ditunjukkan pada Gambar 2.10, propagasi sinyal terhalangi oleh penghalang (bangunan tinggi dan pohon) yang menyebabkan terjadinya refleksi dan difraksi sehingga menyebabkan terjadinya pergeseran fase antara komponen fase utama yang bersuperposisi dengan komponen fase lintasan lain. Hal ini menyebabkan terjadinya delay, redaman amplitude, dan pergeseran fase pada sinyal yang diterima.



Gambar 2.10 Lintasan Propagasi sinyal dari TX ke RX [9]

Additive White Gaussian Noise merupakan model kanal yang sederhana yang mempunyai persamaan *Probability Distribution Function* (PDF) yang ditunjukkan persamaan 2.12 berikut ini:

$$\rho(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left[\frac{-(x-m)^2}{\sigma^2}\right]} \quad (2.12)$$

Berdasarkan Gambar 2.10 output dari antenna array penerima (mengacu pada [9]) dirumuskan sebagai berikut :

$$\begin{aligned} s(t; \theta_l) &= [s_1(t; \theta_l), \dots, s_N(t; \theta_l)]^T \\ &= \alpha_l e^{j\theta} \mathbf{c}_2(\Omega_{2,l}) \mathbf{c}_1(\Omega_{1,l})^T \mathbf{pn}(t - \tau_l) \end{aligned} \quad (2.13)$$

dimana :

$\mathbf{pn}(t - \tau_l)$: deretan pn sekuece termodulasi yang terdelay sebesar τ_l akibat lintasan jamak ke $-l$ dari propagasi dari pemancar ke penerima,

α_l : factor amplitudo kompleks sinyal ke $-l$

\mathbf{c}_2 : *steering vector* dari arah kedatangan (DOA)

\mathbf{c}_1 : *steering vector* dari arah keberangkatan (DOD)

Semua parameter sinyal yang akan diestimasi dinyatakan pada variabel vector sebagai berikut : $\theta_l = [\xi_{1,l} \ \xi_{2,l} \ \tau_l \ \alpha_l]$, karena ada L lintasan yang merupakan komponen lintasan jamak berbeda maka parameter θ_l digabungkan bersamaan menjadi $\theta = [\theta_1, \dots, \theta_L]^T$.

Pada persamaan (2.13) menunjukkan kontribusi ℓ komponen sinyal output lintasan jamak pada antenna array planar, yang ditambahkan dengan noise AWGN yang ditunjukkan dengan persamaan (2.14) berikut ini:

$$\begin{aligned} \mathbf{Y}(t) &= [Y_1(t), \dots, Y_N(t)]^T \\ &= \sum_{l=1}^L s(t; \theta_l) + \sqrt{\frac{N_0}{2}} \mathbf{W}(t) \end{aligned} \quad (2.14)$$

dengan $\mathbf{W}(t)$ menunjukkan *white noise* kompleks dimana $\mathbf{W} \in \mathbb{C}^{N \times 1}$, $\mathbf{Y} \in \mathbb{C}^{N \times 1}$ dan $\mathbf{s} \in \mathbb{C}^{N \times 1}$. Variabel N menunjukkan jumlah elemen di penerima dan variabel M menunjukkan jumlah elemen di pemancar.

1.1.5 Metode Estimasi Parameter Kanal

Metode estimasi kanal dibagi menjadi dua kelompok utama yaitu teknik berbasis spektral (*spectral based*) dan teknik berbasis parameter (*parameter based*). Pada teknik berbasis spektral dibentuk suatu fungsi semacam spektral dengan variabel fungsi adalah parameter yang diestimasi. Puncak fungsi spektrum menunjukkan estimasi parameter tersebut. Teknik spektral dibagi menjadi dua kelompok, yaitu teknik beamforming dan teknik subspace. Contoh teknik beamforming adalah Barlett dan Capon, sedangkan teknik *subspace* adalah *Multiple Signal Classification* (MUSIC) yang memanfaatkan *eigenstructure* dari masukan *covariance* matriks yang diterapkan pada estimasi delay [2].

MUSIC mempunyai kekurangan yang menonjol pada kemampuannya untuk mengestimasi parameter dengan jumlah sinyal mendekati jumlah elemen antenna array, meskipun SNR tinggi. MUSIC hanya mampu mengestimasi parameter-parameter sinyal yang jumlah sinyalnya dibatasi oleh jumlah antenna yang digunakan. Pada [2] menyatakan bahwa kinerja MUSIC semakin memburuk jika diterapkan untuk mengestimasi sinyal lintasan jamak dari hasil pengukuran terutama pada estimasi delay tidak akurat.

Pada teknik berbasis parameter membutuhkan komputasi yang lebih kompleks dan waktu yang lama, dengan hasil estimasi yang lebih akurat. Teknik ini melakukan estimasi dengan pendarian parameter-parameter secara bersama-sama. Contoh dari teknik berbasis parameter adalah *Maximum Likelihood* (ML) yang terbagi menjadi dua golongan, yaitu *Deterministic* ML (DML) yang memodelkan sinyal *deterministic* dan *Stochastic* ML (SML) yang memodelkan sinyal sebagai proses acak Gaussian. Salah satu contoh dari jenis SML adalah SAGE, yang mempunyai sifat konsisten jika estimasi mendekati nilai sebenarnya untuk jumlah data yang tak berhingga, dapat mengestimasi dua sinyal yang koheren, dan dapat diterapkan di geometri sebarang antenna [3].

Teknik gabungan antara metode *subspace* dan *parametric* disebut dengan teknik *parametric subspace based*, yang menjaga kinerja yang baik dengan pengurangan kompleksitas dan waktu komputasi. Contoh dari teknik ini adalah *Estimation of Signal Parameters via Rotational Invariance Techniques* (ESPRIT) yang mempunyai keterbatasan geometri antena array linier dan jumlah sinyal, sedangkan Root-MUSIC hanya diterapkan pada *geometric* array linier.

1.1.6 Algoritma *Space-Alternating Generalized Expectation Maximization* (SAGE).

SAGE merupakan salah satu metode estimasi parameter kanal yaitu delay ($\hat{\tau}_l$), sudut DOA ($\hat{\xi}_{2,l}$) dan sudut DOD ($\hat{\xi}_{1,l}$), dan amplitudo kompleks ($\hat{\alpha}_l$). Parameter tersebut tersusun ke dalam himpunan $\hat{\theta}_l = \{\hat{\tau}_l, \hat{\xi}_{2,l}, \hat{\xi}_{1,l}, \hat{\alpha}_l\}$ dengan $\hat{\xi}_{2,l} = \{\theta_{2,l}, \phi_{2,l}\}$ dan $\hat{\xi}_{1,l} = \{\theta_{1,l}, \phi_{1,l}\}$.

Algoritma SAGE yang digunakan untuk estimasi parameter kanal sinyal lintasan jamak, yang telah digunakan oleh [1] yang bertujuan untuk memperbaiki kelemahan metode *Expectation-Maximization* (EM) dalam hal konvergensinya. Algoritma SAGE banyak diusulkan karena mempunyai kelebihan yang bisa digunakan oleh geometri antena array sebarang dengan jumlah sinyal yang dapat diestimasi parameter tidak dibatasi oleh jumlah elemen antena array yang digunakan serta SAGE mempunyai resolusi tinggi dalam memisahkan sinyal lintasan jamak. Sedangkan kekurangan dari algoritma SAGE adalah komputasinya lebih kompleks.

Algoritma SAGE terdiri dari dua tahap penting yaitu tahap ekspektasi (*Expectation Step*) dan tahap maksimisasi (*Maximization Step*) yang dilakukan satu per satu untuk setiap sinyal yang diestimasi parameter-parameternya. Berikut penjelasan dari *Expectation Step* dan *Maximization Step*.

1.1.6.1 *Expectation Step* (E-Step)

Tahap ekspektasi merupakan proses mendapatkan data dari komponen sinyal tertentu yang terkait langsung dengan parameter-parameternya dengan cara memisahkan satu sinyal dari sinyal-sinyal lain dalam data hasil

observasi (interference cancellation). Berdasarkan [1] metode yang dapat dilakukan pada tahap ekspektasi yaitu *Parallel Interference Cancellation* (PIC) dan *Serial Interference Cancellation* (SIC), dengan menyatakan bahwa metode SIC lebih baik diterapkan pada estimasi parameter sinyal lintasan jamak. Tahap ekspektasi dengan SIC untuk komponen lintasan jamak ke $-l$ pada persamaan 2.15.

$$x_l(t; \widehat{\theta}_{l'}) = y(t) - \sum_{\ell'=1, \ell' \neq \ell}^L s(t; \widehat{\theta}_{l'}) \quad (2.15)$$

dengan $y(t)$ adalah *complete data*, dan $s(t; \widehat{\theta}_{l'})$ menyatakan kontribusi komponen lintasan jamak ke $-l'$ dengan l' adalah indeks komponen-komponen lintasan jamak yang tidak diestimasi.

1.1.6.2 Maximization Step (M-Step)

Tahap berikutnya dari SAGE adalah tahap maksimisasi yang merupakan prose *update* parameter yang dilakukan secara berurutan dengan persamaan sebagai berikut:

$$\widehat{\theta}_{2,l}'' = \arg \max_{\theta_{2,l}} z(\widehat{\phi}_{1,l}', \widehat{\theta}_{1,l}', \widehat{\phi}_{2,l}', \theta_{2,l}, \widehat{\tau}_l''; \widehat{x}_l) \quad (2.17)$$

$$\widehat{\phi}_{2,l}'' = \arg \max_{\phi_{2,l}} z(\widehat{\phi}_{1,l}', \widehat{\theta}_{1,l}', \phi_{2,l}, \widehat{\theta}_{2,l}'', \tau_l''; \widehat{x}_l) \quad (2.18)$$

$$\widehat{\theta}_{1,l}'' = \arg \max_{\theta_{1,l}} z(\widehat{\phi}_{1,l}', \theta_{1,l}, \widehat{\phi}_{2,l}'', \widehat{\theta}_{2,l}'', \widehat{\tau}_l''; \widehat{x}_l) \quad (2.18)$$

$$\widehat{\phi}_{1,l}'' = \arg \max_{\phi_{1,l}} z(\phi_{1,l}, \widehat{\theta}_{1,l}'', \widehat{\phi}_{2,l}'', \widehat{\theta}_{2,l}'', \widehat{\tau}_l''; \widehat{x}_l) \quad (2.19)$$

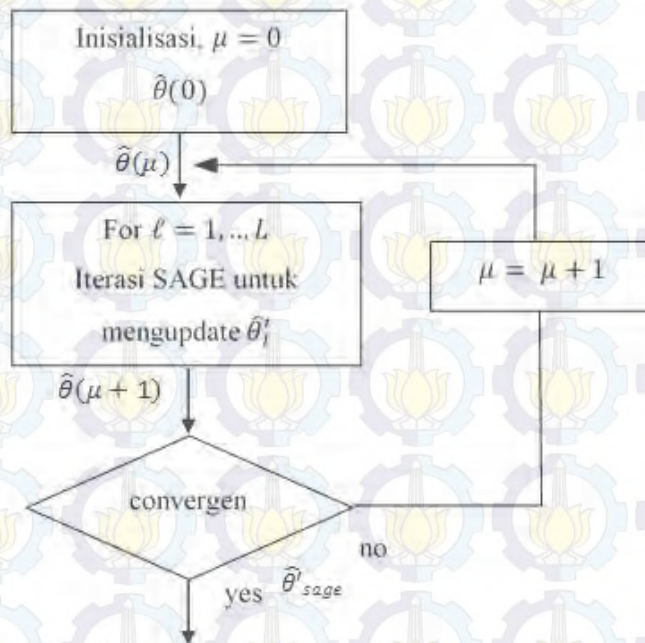
$$\widehat{\alpha}_l'' = \frac{z(\widehat{\phi}_{1,l}'', \widehat{\theta}_{1,l}'', \widehat{\phi}_{2,l}'', \widehat{\theta}_{2,l}'', \widehat{\tau}_l''; \widehat{x}_l)}{[|c_2(\widehat{\phi}_{2,l}'', \widehat{\theta}_{2,l}'')|] [c_1(\widehat{\phi}_{1,l}'', \widehat{\theta}_{1,l}'')] IPT_{sc}} \quad (2.20)$$

Fungsi dari $z(\widehat{\theta}_l; \widehat{x}_l)$ didefinisikan pada persamaan (2.26) berikut ini :

$$z(\widehat{\theta}_l; \widehat{x}_l) \triangleq \widetilde{c}_2(\boldsymbol{\Omega}_{2,l})^H x_l(t; \tau_l) \widetilde{c}_1(\boldsymbol{\Omega}_{1,l})^* \quad (2.21)$$

Dengan $\widetilde{c}_1(\boldsymbol{\Omega}_{1,l}) \triangleq \left| c_1(\boldsymbol{\Omega}_{1,l}) \right|^{-1} c_1(\boldsymbol{\Omega}_{1,l})$ dan $\widetilde{c}_2(\boldsymbol{\Omega}_{1,l}) \triangleq \left| c_2(\boldsymbol{\Omega}_{2,l}) \right|^{-1} c_2(\boldsymbol{\Omega}_{2,l})$ yang merupakan respon normalisasi dari array.

Signal Flow Graphic (SFG) dari SAGE yang menggabungkan dari *Expectation step* dan *Maximization Step* untuk tahap inisialisasi dan estimasi dari algoritma SAGE ditunjukkan pada Gambar 2.11. Gabungan dari E-step dan M-step dikatakan step iterasi SAGE. Siklus iterasi SAGE terdiri dari L step iterasi SAGE, dimana vector parameter dari tiap gelombang diestimasi satu kali [1]. Urutan dari estimasi $\{\hat{\theta}(\mu)\} \triangleq \{\hat{\theta}(\mu): \mu = 0, 1, \dots\}$ secara berulang menyelesaikan EM-step, dimana masing-masing iterasi μ dikenal sebagai pelaksanaan inisialisasi $\hat{\theta}' = \hat{\theta}(\mu)$ dan estimasi $\hat{\theta}_l'' = \hat{\theta}_l'(\mu + 1)$ dengan $\ell = 1, \dots, L$ berdasarkan [1].



Gambar 2.11. *Signal Flow Graph* (SFG) Algoritma SAGE

2.1. 7 Software Defined Radio (SDR)

Software defined radio merupakan komunikasi radio yang terdiri dari beberapa komponen dalam sebuah perangkat keras, misalnya mixer, filter, amplifier, modulator, demodulator, detector dan lainnya, yang diimplimentasikan dengan menggunakan perangkat lunak (software) yang telah diinstall di dalam computer. Pemrosesan sinyal yang telah diatur software yang menggunakan *Field Programmable Gate Arrays* (FPGA), *General Purpose Processors* (GPP) atau perangkat lainnya yang dapat diprogram. Salah satu contoh SDR yang telah

dikembangkan di lab B306 Antenna Propagasi yaitu *Universal Software Radio Peripheral* (USRP) .

2.1.7.1 *Universal Software Radio Peripheral* (USRP)

USRP merupakan salah satu contoh dari SDR dari *National Instrument* dengan jenis yang berbeda. Ada jenis USRP yang bekerja pada frekuensi tertentu dengan menambahkan board tambahan yang dikenal dengan *daughterboard* pada Gambar 2.12 .



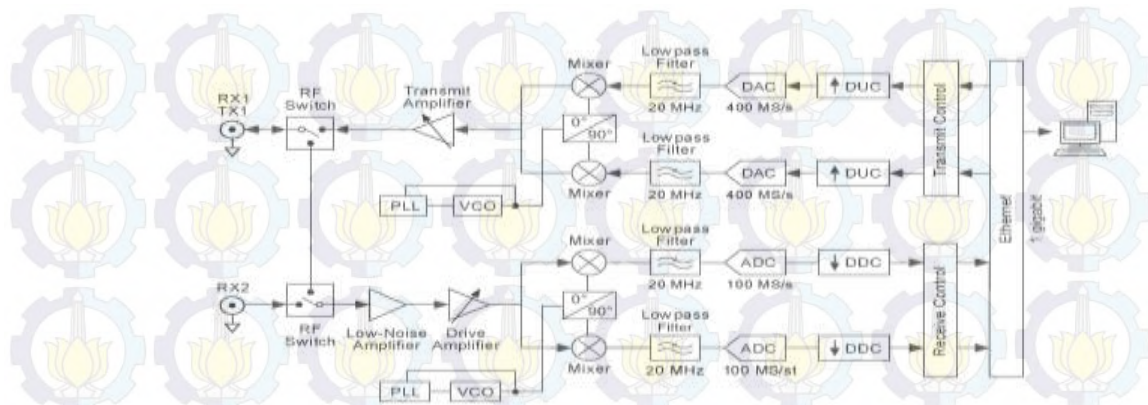
Gambar 2.12 Daughterboard SBX

Daughterboard SBX yang bekerja dengan USRP N2922 adalah daughter SBX dengan spesifikasi pada Tabel 2.2

Tabel 2.2 Spesifikasi Daughterboard

| | |
|---------------|-----------------------------------|
| Frekuensi | 400 MHz – 4 GHz |
| Bandwidth | 40 MHz |
| Channel Count | 1 TX IQ, 1RX IQ, FD (Full Duplex) |

Contoh jenis daughterboard yang digunakan oleh kelompok riset lab antenna propagasi adalah LFTX-RFTX dengan USRP N210 yang digunakan untuk penelitian frekuensi HF, sedangkan pada penelitian ini menggunakan daughterboard SBX dengan USRP N2922 yang bekerja pada frekuensi 2-4 GHz. USRP diintegrasikan dengan menggunakan software Labview dengan men-drug blok device yang tersedia di library Labview, dimana USRP dihubungkan dengan kabel Ethernet yang terkoneksi dengan PC dan labview. Blok diagram keseluruhan dari USRP ditunjukkan pada Gambar 2.13



Gambar 2.13 Blok Diagram USRP N2922

Blok keseluruhan dari USRP yang terdiri dari SBX dan FPGA terintegrasi dengan software labview sehingga parameter-parameter yang dibutuhkan dapat diprogram (dapat diubah-ubah) oleh labview, contoh pada SBX parameter yang bisa diprogram adalah setting antenna, gain tx-rx, LO synthesizer dan filter. Sedangkan pada FPGA yang bisa diprogram adalah parameter attenuasi/gain, *time source*, *internal device time*, frekuensi kerja, *clock source*, few-non signal (IP address, network mass, dan *symbolic name device*). Selain parameter di atas, parameter yang perlu diperhatikan dalam pengiriman sinyal adalah bandwidth sistem seperti *IQ sampling rate* dan *sample width*. IQ sampling rate ditentukan untuk mengetahui batas terkecil untuk mengoperasikan USRP, sedangkan *sample width* berhubungan dengan bandwidth sistem.

Implementasi USRP telah dilakukan oleh kelompok riset lab antenna, yaitu implementasi usrp sebagai radar, MIMO (array antenna), dan komunikasi HF. Implementasi tersebut membutuhkan lebih dari satu perangkat USRP (aplikasi MIMO), dimana antara USRP satu dengan yang lainnya disinkronisasikan menggunakan kabel MIMO, sedangkan sinkronisasi antara USRP sebagai penerima dan pemancar membutuhkan GPS eksternal. Perbedaan tipe USRP N2922 dengan USRP N2910 terletak pada daughterboardnya, kedua USRP tersebut bisa dikonfigurasi secara bersama-sama misalnya dalam komunikasi MIMO menggunakan kabel MIMO seperti Gambar 2.14



Gambar 2.14 Konfigurasi USRP N210 dan N2922 menggunakan kabel MIMO

2.1.7.2 Antena Dual Band

Pada penelitian ini bekerja pada frekuensi wifi, sehingga spesifikasi antena yang diperlukan juga perlu diperhatikan. Penelitian ini bekerja pada frekuensi wifi dengan menggunakan antena dualband jenis monopole omindirectioanl vertikal pada rentang frekuensi 2.4 – 2.5 GHz .



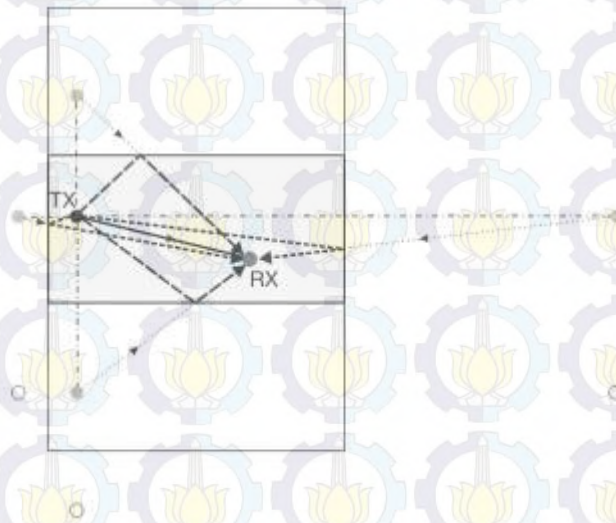
Gambar 2.15 Antena Monopole Dualband

2.1.8 Verifikasi Algoritma SAGE

Verifikasi algoritma dilakukan dengan cara membandingkan keluaran algoritma jika diberi masukan data pengukuran dengan hasil perhitungan ray-tracing yang sesuai dengan pengukuran dengan menggunakan USRP. Ray tracing merupakan salah satu metode model kanal deterministik [10] yang menggambarkan kondisi pengukuran dalam ruang atau di luar ruangan seperti posisi bangunan, posisi ruangan, dan property pemantul seperti dinding yang sangat perlu diperhatikan dalam pengukuran. *Ray tracing* klasik menentukan semua sinyal yang dapat menuju dari satu lokasi TX ke satu lokasi RX. Metode ini terdiri dari 2 langkah, yaitu :

1. Semua komponen sinyal yang datang dari lokasi TX ke lokasi RX ditentukan, yang dilakukan dengan cara menggambarkan jalannya pantulan komponen sinyal. Sinyal yang bisa sampai ke RX akibat refleksi/pencerminan menunjukkan perilaku yang sama seperti sinyal dari sumber virtual yang terletak pada gambar aslinya.
2. Pada step kedua ini, redaman diperhitungkan (karena propagasi ruang bebas dan koefisien refleksi), sehingga dapat mengestimasi semua parameter MPC. Ray Tracing yang umum digunakan adalah Ray tracing 2 dimensi karena tracing 3 dimensi mempunyai tingkat komputasi yang tinggi.

Gambar 2.16 menunjukkan bahwa lingkaran berwarna abu-abu menunjukkan terjadinya refleksi tunggal, lingkaran putih menunjukkan terjadinya refleksi ganda, sedangkan garis titik (*dotlined*) sinar yang berasal dari sumber virtual menuju ke RX, untuk garis putus-putus (*dashed line*) menunjukkan refleksi sebenarnya, dan garis lurus (*solid line*) menunjukkan *line of sight*.



Gambar 2.16. Metode Refleksi pada *Ray Tracing* [11]

Pada [4], verifikasi dilakukan dengan menggunakan ray tracing 2D dengan pendekatan *Geometrical Optics* (GO) yang merupakan metode pencerminan dari posisi TX dan RX pada ruangan pengukuran dimana tata ruang dibuat seperti skala aslinya sehingga mewakili ruangan, posisi TX dan RX sesungguhnya. Dari pencerminan, akan didapatkan *ray launching* (peluncuran sinar) untuk mengoreksi keberangkatan sinyal (DOD) dan juga didapatkan *ray*

tracing (kedatangan sinyal komponen) untuk mengkoreksi kedatangan sinyal (DOA). Sedangkan untuk koreksi delay didapatkan dari panjang lintasan sinyal (d) yang terukur dibagi dengan cepat rambat cahaya ($c = 3 \times 10^8$). Dari pencerminan ini, akan didapatkan semua lintasan yang mungkin dapat diperhitungkan atau yang diabaikan dari lingkungan pengukuran, yang kemudian dari data ada yang dibandingkan dengan hasil estimasi data pengukuran kanal.

2.19 Proses Korelasi untuk mendapatkan Respon Impuls

Untuk mendapatkan respon impuls maka dilakukan korelasi silang antara data yang dikirim dan data yang diterima sesuai dengan persamaan (2.15).

$$\rho_{yx}(\tau) = \sum_{t=-\infty}^{t=\infty} x(t) \cdot y(t - \tau) \quad (2.15)$$

$$\rho_{yx}(\tau) = \sum_{t=-\infty}^{t=\infty} x(t) \cdot x(t - \tau) \cdot h(t - \tau)$$

Dimana :

$x(t)$: input sinyal

$x(t - \tau)$: sinyal input yang telah digeser sejauh (τ)

$h(t - \tau)$: respon impulse yang digeser sejauh (τ)

Dari persamaan (2.15) menunjukkan bahwa untuk mendapatkan respon impuls kanal dapat dilakukan dengan proses korelasi silang antara sinyal $y(t)$ berupa IQ dan sinyal input $x(t)$ berupa deretan pn *sequence* apabila input merupakan auto-korelasi. Fungsi korelasi silang merupakan dua operasi linier sehingga dapat dilakukan tukar posisi dimana auto-korelasi sinyal dilakukan terlebih dahulu kemudian dilakukan konvolusi dengan $h(t)$.

2.20 Threshold Decision [14]

Pengolahan data untuk menghilangkan efek noise dan mendapatkan sinyal asli yang diterima dengan menentukan nilai *threshold*. Langkah awal menentukan *threshold* dengan cara mengestimasi level nilai dari *background thermal noise*.

Noise diasumsikan sebagai Gaussian noise dimana amplitudo noise diasumsikan berdistribusi Rayleigh.

Dalam upaya melakukan mitigasi pada sampel dari sinyal keseluruhan, pada algoritma ini *probability* daya noise secara khusus ditentukan sebesar 5 %. σ_N^2 merupakan median dari sampel dan ς merupakan nilai probabilitas amplitudo dari noise yang melebihi tingkat tertentu didapat dengan menggunakan persamaan (2.16)

$$P = \exp\left(\frac{\varsigma^2}{2\sigma_N^2}\right) \quad (2.16)$$

Dimana :

ς : probabilitas amplitudo dari noise yang melebihi tingkat tertentu

σ_N^2 : median sampel data

Level median yang telah ditentukan (σ_m) dapat ditemukan dengan mengatur $\varsigma = \sigma_m$ dan memasukkan nilai $P = 0.5$ pada persamaan (2.17) sehingga didapatkan :

$$\sigma_N = \frac{\sigma_m}{\sqrt{\ln 4}} \approx 0.85 \sigma_m \quad (2.17)$$

Berbeda dengan algoritma pada umumnya yaitu menetapkan nilai *threshold*, namun pada penelitian ini *threshold* ditentukan terlebih dahulu untuk mendapatkan sebuah nilai *constant false alarm rate* (CFAR). Metode ini sering digunakan untuk mengetahui multipath dengan syarat kebutuhan *threshold* harus ditetapkan sebagai $\varsigma = \eta \sigma_N$ dan ς bernilai konstan. Dari persamaan (2.15) diketahui bahwa probabilitas dari noise yang melebihi batas *threshold* pada sampel yang bukan merupakan sinyal (atau sama dengan nilai probabilitas *false alarm* per sampel) didapatkan dengan persamaan (2.18).

$$P_f = \exp\left(\frac{-\eta^2}{2}\right) \quad (2.18)$$

Dimana P_f merupakan false alarm yang ditentukan 5 %.

Setelah mendapatkan nilai *threshold*, diambil sampel respon impuls yang lebih besar atau sama dengan 5 % dari nilai *threshold*.

Respon impuls yang telah terfilter dengan menggunakan Sousa, diharapkan bebas dari noise, langkah selanjutnya adalah menerapkan konsep tetangga dan konsep konsistensi pada outputan filter Sousa. Tujuan penerapan ini digunakan untuk menentukan jumlah lintasan komponen yang terdeteksi [19].

Konsep cek tetangga dilakukan dengan cara dilakukan pengecekan terhadap setiap puncak impuls yang bersebelahan untuk mendeteksi benar-benar puncak impuls bukan puncak noise. Apabila terdapat impuls yang diapit oleh kedua impuls yang lain, maka impuls tersebut merupakan puncak impuls sinyal, akan tetapi apabila terdapat impuls yang tidak diapit oleh impuls yang lain maka impuls tersebut digolongkan puncak noise.

Setelah melakukan cek tetangga, berikutnya adalah menerapkan cek kemunculan puncak impuls yang konsisten untuk mendapatkan jumlah lintasan komponen, pada [19] menerapkan prosentase kemunculan impuls adalah lebih besar sama dengan 75 %, apabila kemunculan tersebut di bawah 75% maka impuls tersebut dikategorikan sebagai puncak impuls noise.

2.2 Penelitian Sebelumnya

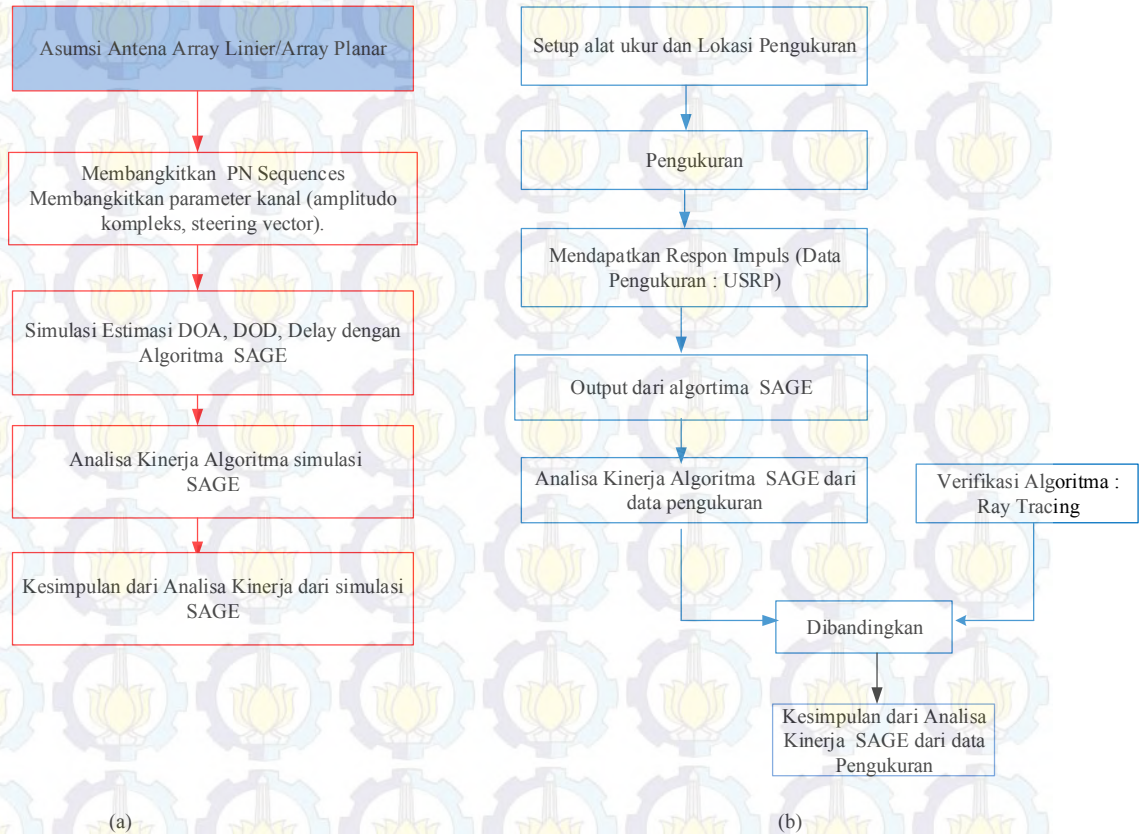
Penelitian mengenai estimasi parameter kanal sudah banyak dilakukan dengan menggunakan bentuk berbagai geometri antena array dengan menerapkan algoritma resolusi tinggi estimasi parameter kanal seperti MUSIC dan SAGE. . P. Handayani dalam tesisnya meneliti tentang karakteristik statistik arah kedatangan komponen-komponen sinyal lintasan jamak pada kanal radio 1,7 GHz di Laboratorium Gelombang Mikro PENS ITS. Penelitian dilakukan melalui pengukuran menggunakan antena array planar sintetik. Estimasi arah kedatangan diperoleh dari pengolahan data hasil pengukuran menggunakan algoritma MUSIC.

Penelitian tersebut dikembangkan lagi dengan menggunakan bentuk geometri array yang lebih kompleks lagi, seperti yang dilakukan [12] yang meneliti penerapan geometri antena array berbentuk kubus menggunakan algoritma MUSIC dan SAGE. Penerapan algoritma MUSIC mempunyai kekurangan yaitu hanya mampu mengestimasi parameter-parameter sinyal yang jumlah sinyalnya dibatasi oleh jumlah antena yang digunakan. Selain itu, penelitian tersebut menggunakan

SAGE untuk mengestimasi parameter kanal (delay, amplitudo kompleks, DOD/ arah keberangkatan sinyal, dan DOA/ arah kedatangan sinyal) yang disebut dengan FD SAGE yang menggunakan antena array kubus sintetis dan verifikasi algoritma dilakukan dengan pengukuran dengan perangkat *Virtual Network Analyzer* di ruangan B306. Penerapan FD-SAGE mempunyai tingkat komputasi yang tinggi, kelemahan inilah yang diperbaiki oleh [3] yang menerapkan IFT FD SAGE yang mengurangi komputasi dari FD SAGE , dimana penelitian ini merumuskan model kanal MIMO-kooperatif dalam bentuk matriks kanal dan mencari karakteristik statistik parameter model.

BAB 3 METODE PENELITIAN

Pada bab ini akan dibahas mengenai metodologi penelitian, yang menggambarkan tahapan penelitian yang dilakukan seperti Gambar 3.1



Gambar 3.1 Diagram Alir Penelitian (a) Simulasi (b) Verifikasi

Berdasarkan Gambar 3.1, tahap penelitian dibagi menjadi dua fase yaitu simulasi dan verifikasi simulasi dari hasil data pengukuran. Simulasi SAGE diterapkan pada antena planar maupun antena linier yang dilakukan dengan menggunakan MATLAB untuk mendapatkan estimasi parameter kanal yang kemudian menggunakan teknik RMSE untuk analisa kinerja SAGE, sedangkan Verifikasi algoritma dilakukan dengan pengukuran kanal real dengan menggunakan perangkat USRP, data yang didapatkan dari pengukuran tersebut diolah dengan SAGE untuk mendapatkan parameter kanal (delay, sudut DOA

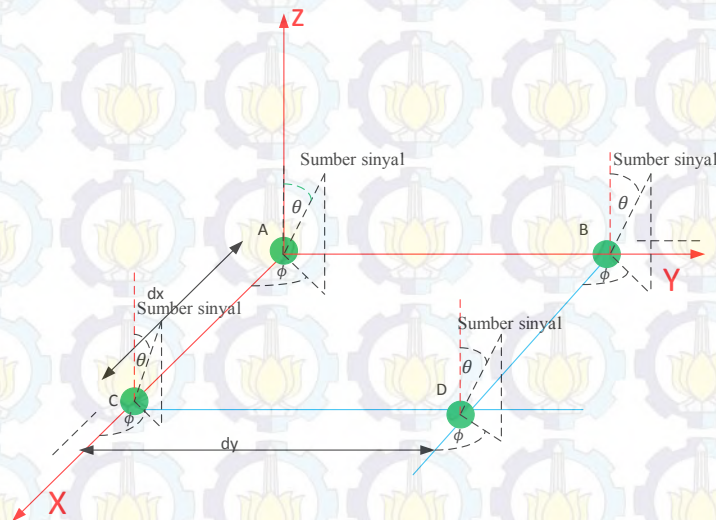
maupun DOD, dan amplitudo kompleks). Hasil dari SAGE akan diverifikasi kembali menggunakan Ray Tracing 2D.

3.1 Asumsi Posisi Antena Array Planar dan Array Linier

Posisi antena array linier diasumsikan terletak pada sumbu z dengan jarak antar elemen antena adalah $\lambda/2$. Penelitian ini berkerja pada frekuensi 2,47 GHz sehingga didapatkan jarak antara elemen antena adalah 6 cm. Antena yang digunakan adalah antena monopole omnidirectional vertical. Posisi elemen antena sangat mempengaruhi hasil estimasi atau hasil validasi dari pengukuran, sehingga dibutuhkan ketelitian dalam meletakkan posisi antena tx dan rx yang digunakan untuk penyusunan matriks respon impuls dari data pengukuran.

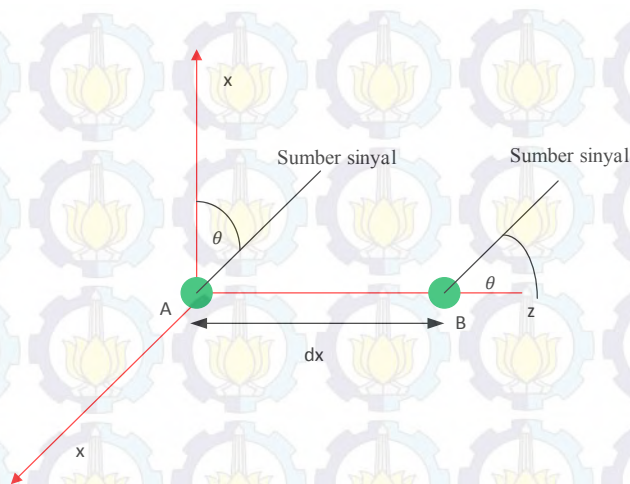
3.1.1 Posisi Antena Array Planar

Elemen antena yang terletak pada bidang xy planar dengan poisisi koordinat xyz dengan titik-titik koordinat setiap elemen adalah $A : [0 \ 0 \ 0]$, $B = [0 \ r \ 0]$, $C = [r \ 0 \ 0]$, $D = [r \ r \ 0]$, untuk r adalah jarak antara elemen antena seperti Gambar 3.2,



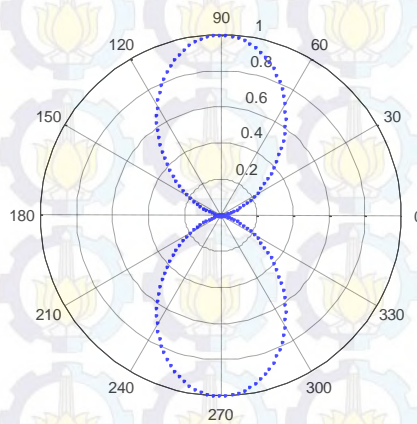
Gambar 3.2 Posisi Elemen Antena Array Planar

Jarak antara elemen antena diatur sebesar $\lambda/2$ ($d_x < \lambda$ dan $d_y < \lambda$) untuk menghindari terjadinya *grating lobe* baik pada antena array planar dan antena linier. Validasi SAGE menggunakan antena array linier dengan posisi seperti gambar 3.3



Gambar 3.3 Posisi Elemen Antena Array Linier

Posisi elemen A terletak pada $[0,0,0]$ dalam koordinat xyz, sedangkan elemen B terletak pada $[0,0,r]$. Pola radiasi dua dimensi untuk antena array linier terlihat pada Gambar 3.4, pola radiasi ini sangat penting digunakan untuk mengetahui kinerja SAGE dalam mengestimasi arah datang maupun arah berangkat sinyal.



Gambar 3.4 Pola Radiasi Antena Array Linier

Pola radiasi tersebut menunjukkan bahwa sudut elevasi yang menuju ke 0° dan 180° mengalami pelemahan gain apabila terdapat komponen sinyal yang datang atau berangkat pada arah elevasi tersebut, akan mengalami keambiguan. Pola radiasi tersebut merupakan hasil perkalian array factor untuk antena linier dengan pola elemen dipole.

3.2 Pembangkitan PN Sequences dan Parameter Kanal

Model sinyal diasumsikan seperti persamaan (2.13) dan Gambar 2.10 merupakan gambar dari lintasan sinyal propagasi yang bergerak dari pemancar ke penerima dengan halangan yang berada di lingkungan, sehingga menyebabkan refleksi, difraksi, dan refraksi dari sinyal yang melalui beberapa lintasan yang berbeda (1,2,...L). Hal ini membuat sinyal yang melewati lintasan akan terdelay, amplitudo diredam dan fase akan berubah.

Sinyal yang dikirim $p_n(t)$ berupa deretan pn sequence dan data yang diterima (data IQ) dalam domain waktu, hal ini menunjukkan bahwa crosscorelasi antara data IQ yang diterima dengan sinyal $p_n(t)$ menghasilkan bentuk respon impuls. Pn sequence yang dibangkitkan dengan orde 8, dengan panjang dari pn sequence tersebut adalah 510 bit, maka dimensi sinyal yang dibangkitkan adalah $L \times K$. L menunjukkan jumlah komponen yang terdeteksi di penerima dan K menunjukkan panjang pn sekuen. Sinyal yang tiba di penerima diasumsikan melewati lintasan jamak (*multipath*) sehingga mendapatkan parameter kanal delay, sudut (arah datang dan arah berangkat), dan amplitudo kompleks ($\theta_l = \hat{\tau}'_l, \hat{\zeta}_{2,l}', \hat{\zeta}_{1,l}', \hat{\alpha}_l$). Parameter tersebut dibangkitkan secara acak.

- Pembangkitan delay : delay yang dibangkitkan acak sebanyak deretan pn sequence, misalkan pada penelitian ini pn sequence yang dibangkitkan sebanyak 510 sampel, maka delay dibangkitkan antara 0 sampai 509 ns baik untuk pendeteksian satu sinyal atau lebih, dan dimensi delay yang dibangkitkan adalah $L \times 1$.

- Arah Kedatangan dan Keberangkatan Sinyal

Pembangkitan arah datang (DOA) dan arah berangkat (DOD) dari suatu sinyal berhubungan dengan posisi antenna, jarak elemen antar antenna, dan frekuensi yang digunakan untuk pembentukan matriks *steering vector*. Array linier mempunyai pola radiasi hanya pada arah elevasi ($0-\pi$) dengan resolusi 1° , maka pembentukan steering vector untuk doa dan dod dari antenna array linier berdimensi 2×181 , dimana 2 menunjukkan jumlah antenna penerima/pemancar. Dengan kata lain, steering vector tersebut berdimensi $M \times 181$. Steering vector

digunakan untuk scanning sudut dari sinyal yang diterima, untuk mendapatkan estimasi arah datang dan arah berangkat dari sinyal.

Sedangkan steering vector array planar terletak pada bidang xy yang mengcover sudut elevasi ($0-\pi$) dan sudut azimuth ($0-2\pi$) dengan dimensinya $M \times (181^\circ \times 361^\circ)$. Namun dimensi arah datang dan arah berangkat sinyal input adalah $L \times 1$.

- Amplitudo Kompleks

Amplitudo kompleks dibangkitkan secara random dengan varian 1 dan mean nol. Dimensi amplitudo kompleks yang dibangkitkan berdimensi $L \times 1$.

Parameter kanal tersebut dikalikan dengan deretan pn sehingga membentuk sinyal yang diterima (sesuai persamaan 2.13) dengan ukuran $M \times K$, dimana M menunjukkan jumlah antenna dan K menunjukkan banyaknya deretan bit yang dibangkitkan. Pn sequence yang telah dibangkitkan ditambahkan phase random sebagai akibat difraksi atau pantulan dari suatu sinyal yang terkirim. Selanjutnya sinyal tersebut ditambahkan noise, sehingga seolah-olah pemrosesan sinyal yang dilakukan oleh SAGE adalah sinyal di penerima.

3.3 Simulasi Estimasi DOA, DOD, Delay, dan Amplitudo Kompleks

Simulasi untuk mendapatkan parameter kanal delay, sudut, dan amplitudo kompleks menggunakan SAGE, tujuan penelitian ini adalah mengestimasi parameter sudut (arah datang dan arah berangkat sinyal), delay, dan amplitudo kompleks. Hal ini bertujuan agar dua sinyal yang datang bersamaan pada arah yang sama, tetapi melalui lintasan yang berbeda, dapat terdeteksi oleh SAGE. Tujuan tersebut dapat diterapkan pada implementasi USRP yang mempunyai resolusi delay yang rendah khususnya untuk pengukuran realtime.

Respon impuls $H(k)$ yang akan diestimasi oleh SAGE, merupakan data IQ atau bilangan kompleks. Tahap-tahap estimasi tersebut, dimulai dari pembangkitan sinyal yang diterima dengan mengetahui inputan parameter kanal seperti delay, sudut, dan amplitudo kompleks dengan harapan bahwa hasil output SAGE mendekati dengan inputan yang diberikan.

Algoritma SAGE terdiri dari dua konsep dasar yaitu *unobservable admissible data* $X(k)$ dan *observable incomplete data* $H(k)$. Dekomposisi sinyal $X(k)$ diproses dua langkah, yakni *Expectation* (E) Step yang menghitung nilai yang diharapkan dari satu lintasan pada output antenna penerima, sedangkan *Maximization* (M) Step, fungsi kemungkinan (*likelihood function*) adalah proses memaksimalkan parameter dari satu lintasan secara sekuensial (berurutan), dimana urutan tersebut mewakili tahapan inisialisasi dan iterasi. Iterasi tersebut dilakukan untuk semua lintasan, akan berhenti sampai semua parameter mencapai konvergen. Secara garis besar, parameter yang didapatkan dari SAGE ditunjukkan pada flowchart Gambar 3.5.

- Tahap Inisialisasi SAGE

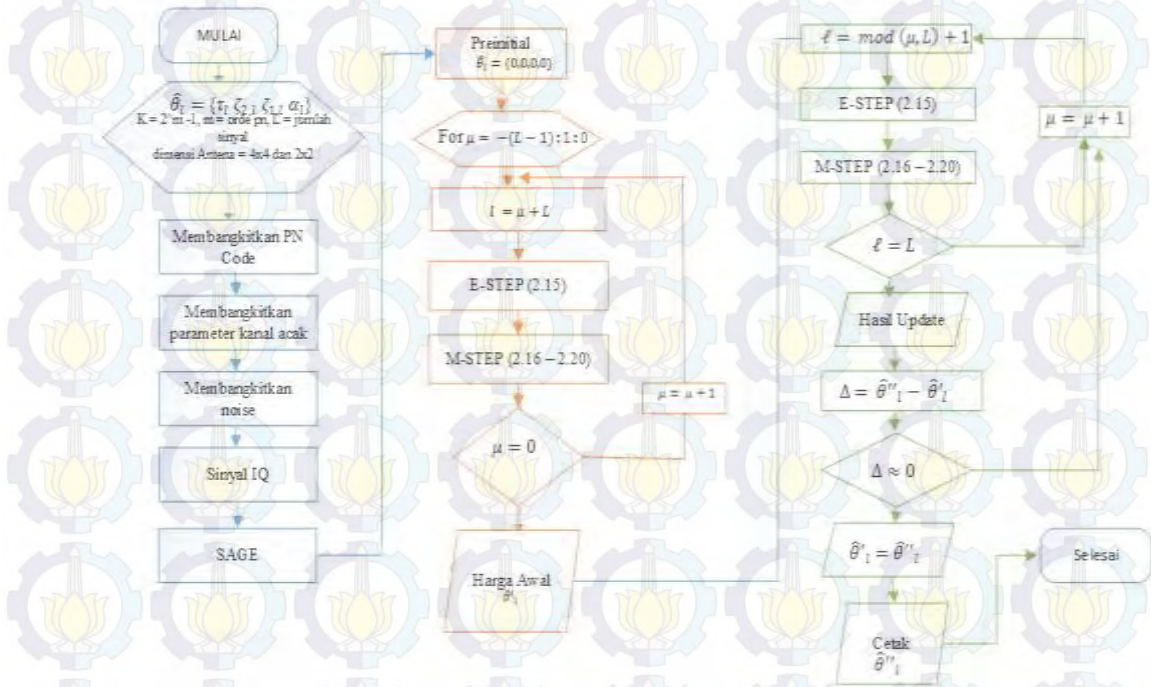
Tahap inisialisasi ini dimulai dengan pre-inisialisasi setiap parameter kanal, diberi nilai sama dengan nol ($\theta = \{0,0,0,0\}$), beda halnya apabila SAGE diterapkan pada data hasil pengukuran, dimana delay diinisialisasikan dari hasil crosscorelasi antara data yang dikirim dan data yang diterima. Pada sub bab ini dijelaskan mengenai langkah-langkah SAGE secara simulasi untuk mendapatkan estimasi sudut ($\widehat{\zeta_{2,l}'}, \widehat{\zeta_{1,l}'}$), delay ($\widehat{\tau}_l'$) dan amplitudo kompleks ($\widehat{\alpha}_l'$).

- Tahap Update SAGE

Tahap update SAGE atau disebut sebagai tahap iterasi SAGE, dilakukan dengan cara meng-update nilai parameter yang didapatkan pada tahap inisialisasi untuk mendapatkan nilai parameter terbaru sampai mencapai konvergen, dimana variabel konvergen perlu diinisialisasi di awal sebelum tahap ini. Pada simulasi, variabel konvergen diinisialisasi sama dengan nol, yang artinya input parameter kanal dengan hasil estimasi parameter kanal sama.

Proses algoritma SAGE yang dipaparkan oleh diagram flowchart Gambar 3.5, sangat kompleks yang dimulai dari proses pembangkitan sinyal yang akan diestimasi, pembangkitan parameter kanal (ditandai warna biru), inisialisasi (ditandai dengan warna orange) sampai tahap update atau iterasi (ditandai warna

hijau). Pada tahap inialisasi dan update (m-step) dilakukan estimasi sudut , delay, dan amplitudo kompleks yang dirumuskan pada persamaan (3.1) – (3.4)



Gambar 3.5 . Flowchart Simulasi SAGE

- Estimasi Sudut untuk arah kedatangan sinyal dan arah keberangkatan sinyal. Pada array planar arah datang dan arah berangkat sinyal terdiri dari sudut elevasi dan sudut azimuth. Sedangkan pada array linier arah datang dan arah berangkat sinyal terdiri dari elevasi saja yang dituliskan pada persamaan (3.1) – (3.5). Persamaan tersebut merupakan persamaan dalam simulasi SAGE untuk pendeteksian parameter dimulai dari parameter spasial.

1. Inisialisasi DOA $(\widehat{\zeta_{2,l}}')$

$$\widehat{\zeta_{2,l}}'' = \arg \max_{\zeta_{2,l}} \{z_l(\zeta_{2,l}, \widehat{\zeta_{1,l}}', \tau_l'; \hat{x}_l(k; \hat{\theta}_l'))\} \quad (3.1)$$

Persamaan (3.1) diuraikan menjadi persamaan 3.2

$$z_l(\zeta_{2,l}) = \arg \max_{\zeta_{2,l}} \left\{ \sum_{m=1}^M \left| \sum_{k=1}^K c_2(\Omega_{2,l})^H \hat{x}_{l,m,k}(k; \hat{\theta}_l') \right| \right\} \quad (3.2)$$

Dimana :

- $\widehat{\zeta_{1,l}}'$: parameter DOD yang diberi nilai nol
 $\widehat{\tau_{1,l}}$: parameter delay yang diberi nilai nol
 $\hat{x}_l(t; \widehat{\theta}'_l)$: output ekspektasi step (dimensi N x K)
 $\mathbf{c}_2(\Omega_{2,l})$: matrik *steering vector* DOA (N x 180*360)
M : jumlah antenna penerima
K : sample pengamatan

Pada persamaan (3.1) merupakan persamaan inisialisasi dari parameter sudut (arah datang sinyal), yang terdiri dari input parameter sudut DOD ,delay, steering vector, dan output tahap E-Step. Nilai input parameter sudut didapatkan dari preinisialisasi (diberi nilai nol).

2. Inisialisasi DOD ($\widehat{\zeta_{1,l}}'$)

Setelah mendapatkan nilai parameter DOA dari tahap 1, langkah berikutnya adalah menghitung parameter DOD dengan persamaan 3.3.

$$\widehat{\zeta_{1,l}}'' = \arg \max_{\zeta_{1,l}} \{ z_l(\zeta_{1,l}, \widehat{\zeta_{2,l}}'', \widehat{\tau}_l'; \hat{x}_l(k; \widehat{\theta}'_l)) \} \quad (3.3)$$

Persamaan (3.3) diuraikan pada persamaan (3.4),

$$z_l(\zeta_{1,l}, \widehat{\zeta_{2,l}}'', \widehat{\tau}_l'; \hat{x}_l(k; \widehat{\theta}'_l)) = \arg \max_{\zeta_{1,l}} \left\{ \sum_{n=1}^N \left| \sum_{k=1}^K \mathbf{c}_2(\Omega_{2,l})^H \hat{x}_{l,n,k}(k; \widehat{\theta}'_l) \mathbf{c}_1^*(\Omega_1) \right| \right\} \quad (3.4)$$

Dimana :

- $\mathbf{c}_2(\Omega_{2,l})^H$: Steering vector dari sudut DOA yang sudah terupdate (N x 1), operator H menunjukkan transpose conjugate.
 $\mathbf{c}_1^*(\Omega_1)$: matriks steering vector dod (M x (180*360))
N, M : jumlah antenna elemen pemancar dan penerima.

Selanjutnya, inisialisasi parameter DOD didapatkan dari inputan parameter : DOA yang telah terinisialisasi (dari proses ke satu), steering vector yang dibangkitkan acak, dan output E-step.

3. Inisialisasi Delay (τ_l'')

Inisialisasi delay didapatkan dari tahap inisialisasi doa dan dod di atas, sehingga persamaan untuk inisialisasi delay ditunjukkan pada persamaan (3.5).

$$\widehat{\tau}_l'' = \underset{\tau_l}{\arg \max} \{z_l(\widehat{\zeta}_{1,l}'', \widehat{\zeta}_{2,l}'', \tau_l; \widehat{x}_l(k; \widehat{\theta}_l))\} \quad (3.5)$$

Fungsi cost pada persamaan (3.5), dijabarkan pada persamaan (3.6),

$$z_l(\tau_l, \widehat{\zeta}_{2,l}'', \widehat{\zeta}_{1,l}'', \widehat{x}_l(k; \widehat{\theta}_l)) = \underset{\tau_l}{\arg \max} \sum_{k=1}^K \left| \sum_{m,n=1}^{M,N} \mathbf{c}_2(\Omega_{2,l})^H \widehat{x}_{l,m,n,k}(k; \widehat{\theta}_l) \mathbf{c}_1^*(\Omega_{1,l}) \mathbf{p}n(k - \tau_l) \right| \quad (3.6)$$

Dimana :

- $\mathbf{c}_1^*(\Omega_{1,l})$: Steering vector dari sudut DOD yang sudah terupdate (M x 1), operator * menunjukkan conjugate.
- $\mathbf{c}_2(\Omega_{2,l})^H$: Steering vector dari sudut DOA yang sudah terupdate (N x 1), operator H menunjukkan transpose conjugate.
- $\mathbf{p}n(t - \tau_l)$: Deretan pn sekuen yang terdelay τ_l

Inisialisasi dari delay didapatkan dari inputan parameter : DOA dan DOD yang telah terinisialisasi (dari proses ke 2 dan 3), output ekspektasi, dan sinyal pn(t) yang terdelay setiap satu sample (1ns) sebanyak 510 sample.

4. Inisialisasi Amplitudo Kompleks

Inisialisasi dari persamaan amplitudo kompleks tergantung dari parameter delay, DOA, dan DOD, yang ditunjukkan pada persamaan (3.7),

$$\widehat{\alpha}_l'' = \frac{1}{KNM} \{z_l(\widehat{\zeta}_{1,l}'', \widehat{\zeta}_{2,l}'', \widehat{\tau}_l''; \widehat{x}_l(k; \widehat{\theta}_l))\} \quad (3.7)$$

Persamaan (3.7) inialisasi dari amplitudo kompleks tergantung dari proses inialisasi DOA, DOD, dan delay (dari proses 1, 2, dan 3).

Persamaan (3.1) – (3.7) merupakan persamaan fungsi cost yang maksimal untuk mendapatkan parameter estimasi sudut, estimasi delay dengan menggunakan hasil estimasi sudut yang baru didapatkan, begitu halnya untuk estimasi amplitudo kompleks menggunakan hasil estimasi sudut dan delay yang baru diperoleh. Hal ini terus menerus dilakukan (sekuensial) sampai tahap iterasi atau update untuk mendapatkan nilai yang konvergen.

3.4 Hasil Simulasi

Setelah membangkitkan parameter-parameter simulasi, khusus untuk parameter kanal (DOA, DOD, delay dan amplitudo kompleks) yang diproses dengan SAGE mencapai konvergen dimana output dari SAGE mendekati estimasi dari parameter kanal tersebut. Keakuratan hasil simulasi dapat diuji dengan menggunakan metode *Root Mean Square Error* (RMSE), pengujian dilakukan dengan membandingkan parameter kanal input dengan hasil estimasi parameter kanal. Semakin kecil error yang terjadi, semakin akurat estimasi yang dilakukan. Keakuratan tersebut akan diuji dengan variasi parameter SNR, variasi jumlah sinyal, dan perubahan sudut elevasi, hal ini dilakukan untuk mengetahui kinerja dari algoritma SAGE dalam mengestimasi parameter kanal. RMSE dapat dirumuskan sebagai berikut pada persamaan (3.8),

$$RMSE = \frac{1}{I} \frac{1}{L} \sqrt{\sum_{i=1}^I \sum_{l=1}^L (\widehat{\theta}_{l,i} - \theta_{l,i})^2} \quad (3.8)$$

dimana $\widehat{\theta}_{l,i}$ adalah hasil estimasi parameter kanal, sedangkan $\theta_{l,i}$ merupakan parameter kanal yang dibangkitkan atau yang diinputkan, indeks ke l adalah jumlah multipath sinyal dengan $l = 1, 2, \dots, L$ dan i adalah jumlah perulangan (iterasi) dengan $i = 1, 2, \dots, I$. Hasil estimasi parameter kanal dengan menggunakan RMSE yang menghitung hasil estimasi SAGE dengan hasil parameter input yang

dibangkitkan secara acak setiap satu kali iterasi, sehingga dalam perhitungannya dibagikan dengan banyak iterasi yang digunakan, pada penelitian ini iterasi yang digunakan 10 kali iterasi.

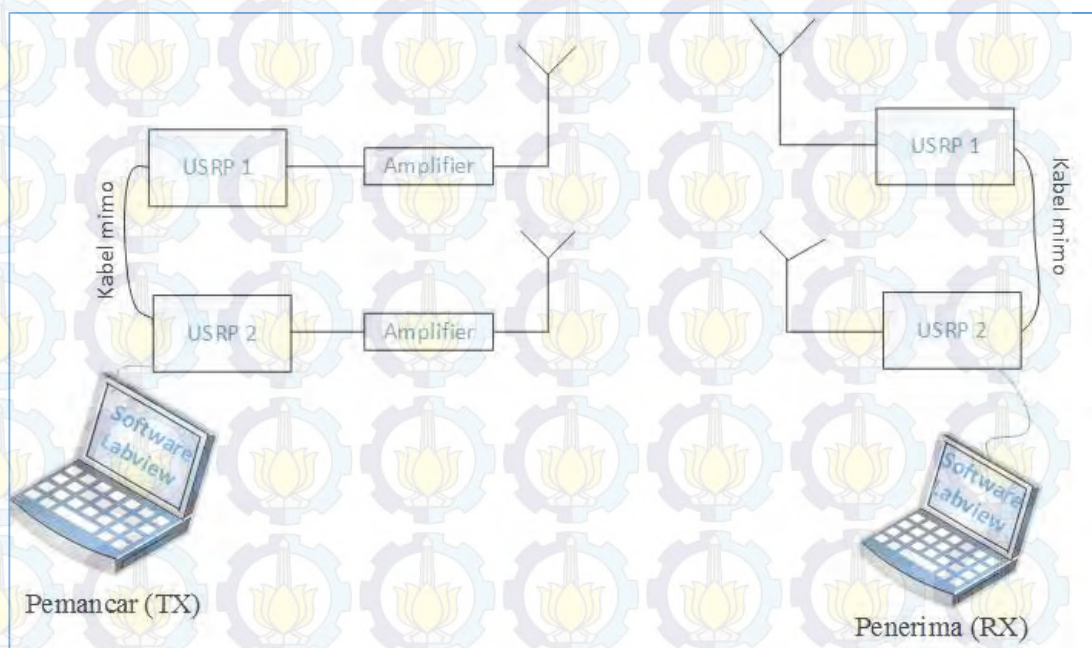
Perhitungan RMSE untuk kinerja SAGE perlu diketahui untuk antenna linier maupun antenna array planar. Kinerja SAGE antenna array linier perlu dianalisa dengan detail agar komponen sinyal yang terdeteksi dari data pengukuran dapat teranalisa dengan baik yang sesuai dengan kinerja SAGE.

3.5 Pengukuran Respon Kanal Untuk Validasi Algoritma SAGE

Setelah kinerja SAGE pada antenna array linier dan planar diketahui, maka langkah selanjutnya adalah membuktikan kinerja SAGE dengan menggunakan data pengukuran. Data pengukuran tersebut berupa data IQ, yang didapatkan dari perangkat USRP. Ketersediaan peralatan USRP di Laboratorium B306 Antena dan Propagasi Jurusan Teknik Elektro ITS terbatas untuk penerapan antenna array planar, sehingga pengukuran respon impuls menggunakan 4 perangkat USRP yaitu 2 USRP sebagai pemancar dan 2 USRP yang lainnya sebagai penerima. USRP yang digunakan adalah tipe N2922, sedangkan 3 lainnya USRP jenis N210. Dari keempat USRP tersebut menggunakan daughterboard yang sama yaitu SBX.

Pengukuran dilakukan di sekitar frekuensi wifi (2.4 GHz), sehingga sebelum pengukuran, dilakukan pengecekan kanal frekuensi yang kosong pada lokasi pengukuran dengan menggunakan aplikasi wifi analyzer, agar tidak berinterferensi dengan frekuensi wifi lain yang sedang digunakan. Setup pengukuran ditunjukkan pada Gambar 3.6, dari blok diagram tersebut satu pc pemancar atau penerima terhubung langsung dengan satu USRP menggunakan kabel Ethernet dan sinkronisasi antar dua USRP tersebut menggunakan kabel MIMO. USRP terhubung ke amplifier dengan menggunakan kabel jumper, sedangkan amplifier ke antenna pemancar menggunakan konektor male to male.

USRP terhubung langsung dengan pc/laptop yang dioperasikan oleh software labview. Software labview yang terinstall pada pc/laptop diprogram untuk mengirimkan deretan pn sequence (pada sisi pemancar) dan menerima sinyal baseband (pada sisi penerima). Sinyal baseband dalam bentuk kompleks akan diolah dan diproses dengan SAGE untuk mendapatkan parameter kanal.



Gambar 3.6 Blok Diagram Pengukuran

Daya terima USRP sekitar 0 dBm sampai -100dB, tapi setelah dilakukan pengukuran dengan jarak melebihi 15 m, daya yang diterima oleh USRP adalah di bawah -100 dBm atau sinyal noise yang diterima, sehingga dibutuhkan amplifiier wifi yang berkerja pada frekuensi 2.4-2.5 GHz pada sisi pemancar, untuk memperkuat sinyal dipancarkan. Pada USRP terdapat parameter yang perlu diinputkan, yang mempengaruhi pengukuran seperti pada sisi pemancar terdiri dari IQ rate, *sample width*, orde pn sequence yang dibangkitkan, gain tx, sedangkan pada sisi penerima parameter yang perlu diinputkan terdiri dari panjang frame dan gain rx .

3.6 Setup Pengukuran

Pengukuran dilakukan pada frekuensi 2450MHz di ruang indoor Hall Gedung Baru Politeknik Elektronika Negeri Surabaya , dengan panjang orde pn yang dikirimkan adalah 8 atau panjang code yang dikirimkan adalah 510. Tabel 1 menunjukkan parameter set up pengukuran.

Tahap berikutnya adalah dengan tahap kalibrasi USRP, dimana kalibrasi bertujuan untuk mengetahui level sinyal di penerima. Kalibrasi dilakukan dengan

mengukur daya level sinyal di penerima sebagai fungsi jarak, hingga terdeteksi jarak maksimal yang bisa dijangkau oleh peralatan tersebut.

Validasi pengukuran dilakukan dengan cara ray tracing 2 dimensi, dimana ray tracing menggambarkan penjajakan sinyal dengan menarik garis lintasan yang memungkinkan terjadinya pantulan akibat dinding atau penghalang lainnya yang ada pada tata ruang Hall PENS. Komponen sinyal LOS dan komponen satu pantulan yang digunakan sebagai validasi dari kinerja SAGE.

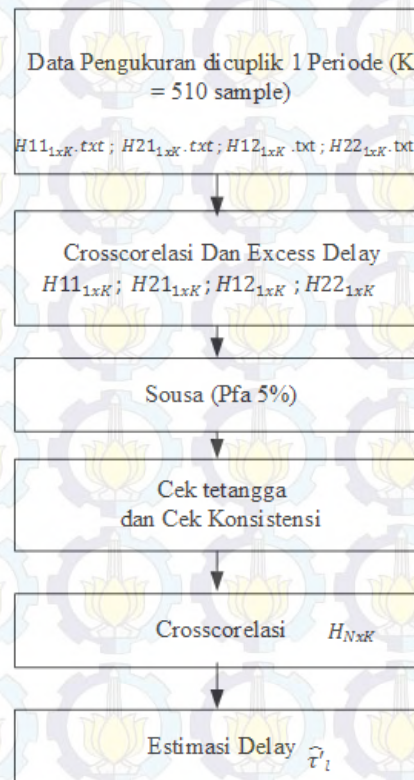
Tabel 3.1 Setup Pengukuran

| | |
|---------------------------|--------------------------|
| Frekuensi | 2450 MHz |
| Daya Terima USRP | 0 dBm : (- 100dBm) |
| Gain Transmit Amplifier | 17 dB |
| Sample width | 8 bit |
| Gain internal USRP | 31.5 dB |
| Daya Pancar USRP | 20 dBm |
| Orde PN | 8 |
| Jumlah Sample | 510 |
| IQ Rate | 33 MHz |
| Chiprate | 16,5MS/s |
| Tc(periode chip) | 60ns |
| Resolusi Temporal Sounder | 60ns ($\geq T_c$) |
| Panjang Respon Maksimal | 15300 ns |
| Tinggi Antena | 20 cm |
| Jenis Antena | Monopole Omnidirectional |
| Frekuensi antena | Vertical |
| | 2.4 -2.48 GHz |

3.7 Pengolahan Data

Metode pengukuran menggunakan TDMA (*Time Division Multiplexing Access*), pengiriman deretan PN sequence dilakukan secara bergantian untuk setiap pemancar yang berbeda, akan tetapi diterima oleh 2 penerima secara bersamaan , sehingga data yang terekam pada sisi penerima perlu diperhatikan untuk penamaanya. Hal ini dapat mempengaruhi penyusunan matriks kanal H, yang akan diproses di dalam SAGE.

Penelitian ini bergabung dengan penelitian yang dilakukan oleh Martianda Erste Anggraeni dengan topik karakteristik penghambur local, dimana data pengukuran digunakan bersama-sama. Penelitian ini mengambil sample data pengukuran Hall, PENS sebagai verifikasi kinerja SAGE. Flowchart pengolahan data ditunjukkan pada Gambar 3.7



Gambar 3.7 Proses Pengolahan Data

Data pengukuran yang terekam oleh penerima tersimpan di Diadem (media penyimpanan pada Labview) terdiri dari beberapa periode, setiap periode terdiri dari 510 sample data yang terekam. Data pengukuran tersebut dicuplik satu periode dari sekian banyak periode yang terekam dan diubah ke dalam bentuk file.txt, hal ini dilakukan agar mengurangi komputasi pembacaan data dalam diadem oleh matlab. Dimensi data pengukuran yang akan diproses oleh SAGE adalah 4×510 , yang tersusun dalam matriks sebagai berikut : $H = [H11; H21; H12; H22]$, dimana setiap data tersebut berukuran 1×510 . File data H11.txt terekam dari pemancar ke satu dan diterima oleh penerima ke satu, sedangkan file data H21.txt terekam dari pemancar satu ke penerima dua, file H12.txt terekam dari pemancar dua ke penerima satu, dan file H22.txt terekam dari pemancar dua ke penerima dua (seperti

tabel 3.2). Sedangkan data deretan pn sequence yang dikirim terekam dalam file tdms, yang kemudian disimpan ke dalam .txt, dimensi data yang dikirim ini adalah 1x510.

Tabel 3.2 Urutan Penyimpanan Data Pengukuran

| Waktu Transmit | Posisi Tx | Posisi Rx | Nama File |
|----------------|-----------|-----------|-----------|
| 1 | A | A | H11.txt |
| | | B | H21.txt |
| 2 | B | A | H12.txt |
| | | B | H22.txt |

Posisi A dan B menunjukkan posisi dari antenna pemancar dan penerima seperti gambar 3.3. Setelah data yang diterima sesuai dengan penamaan nama file, langkah berikutnya adalah mengkorelasikan data yang diterima dengan data yang dikirim. Gambar 3.8 menunjukkan salah satu respon impuls yang merupakan hasil crosscorelasi dari data pengukuran, dimana magnitudo sinyal tertinggi digeser ke titik nol. Persamaan korelasi silang ditunjukkan pada persamaan 3.1 berikut :

$$\rho_{yx}(\tau) = \sum_{t=-\infty}^{t=\infty} x(t).x(t-\tau).h(t-\tau) \quad (3.1)$$

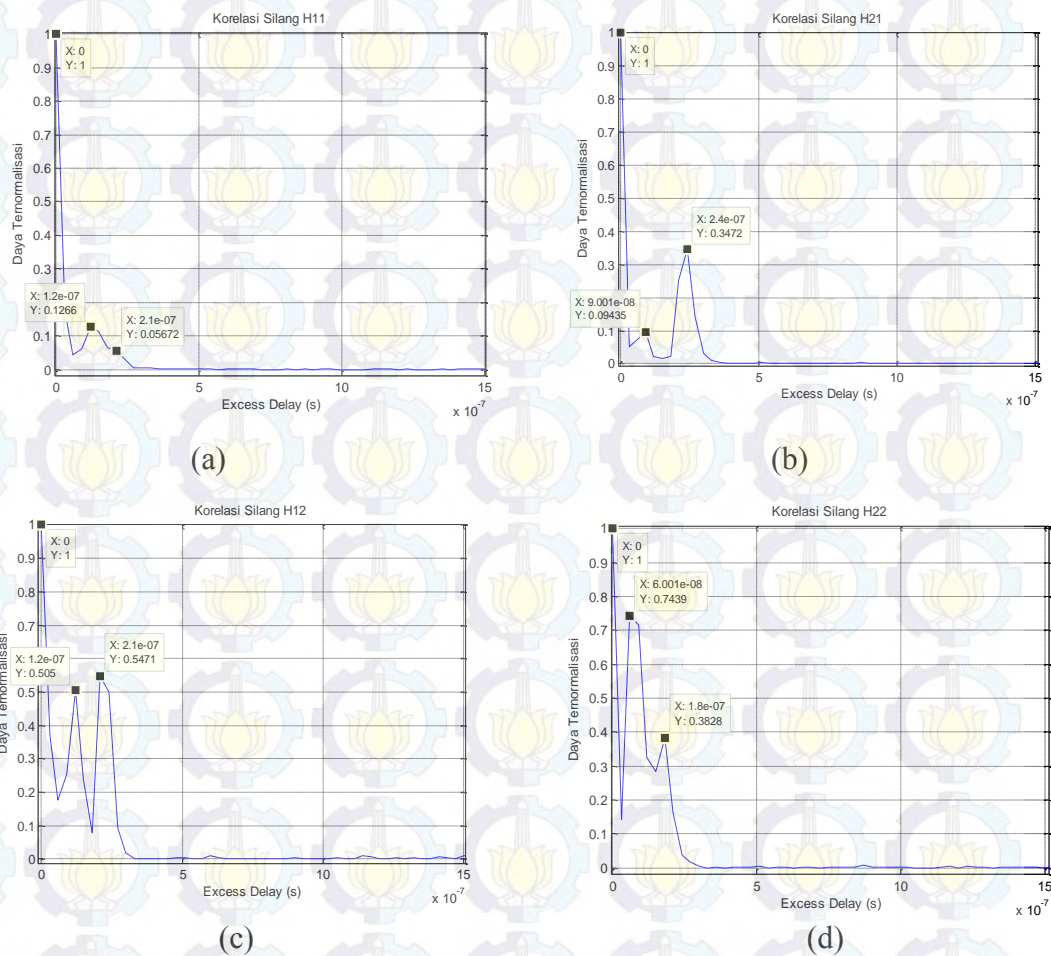
Dimana :

$x(t)$: input sinyal

$x(t-\tau)$: sinyal input yang telah digeser sejauh (τ)

$h(t-\tau)$: respon impulse yang digeser sejauh (τ)

Hasil crosscorelasi dari masing –masing data yang diterima dengan data yang dikirim, dimana puncak tertinggi diasumsikan sinyal LOS dan digeser ke titik nol (excess delay) ditunjukkan pada Gambar 3.8, Penggeseran tersebut dilakukan karena perangkat USRP antara TX dan RX selama pentransmisian data tidak sinkron, yang akan berpengaruh terhadap pendeteksian parameter kanal delay. Respon impuls H11, H21, H12, dan H22 bersifat periodik untuk semua data yang terekam di diadem.



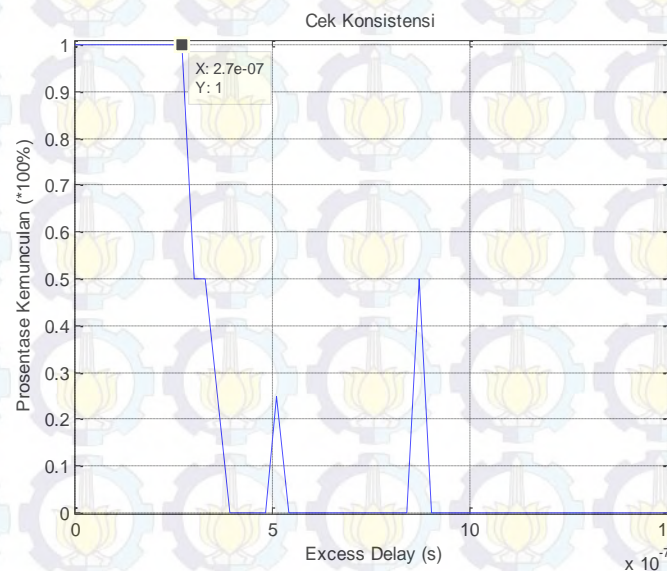
Gambar 3.8 Respon impuls (a) H11 (b) H21 (c) H12 (d) H22

Selanjutnya, dari keempat respon impuls tersebut diterapkan konsep souse. Konsep sousa ini digunakan untuk menghilangkan efek noise dan mendapatkan sinyal asli. Probabilitas daya noise dari keseluruhan sinyal ditetapkan 5% persen untuk mendapatkan sinyal aslinya. Tahap berikutnya adalah menggunakan konsep tetangga dan konsep konsistensi. Konsep tetangga dan konsistensi ini digunakan untuk menetapkan apakah suatu puncak (peak) dalam respon impuls benar-benar sebuah lintasan jamak dan bukan puncak noise.

Metode dari pengecekan konsep tetangga dan konsistensi adalah dari respon impuls yang diterima (seperti Gambar 3.8) adalah sebagai berikut, Setiap respon impuls terdiri dari 510 sample nilai impuls, setiap sample tersebut dilakukan pengecekan terhadap impuls sample ke-n dengan impuls sample ke n+1 dan impuls sample ke n-1, jika impuls sample ke n tersebut diapit oleh dua

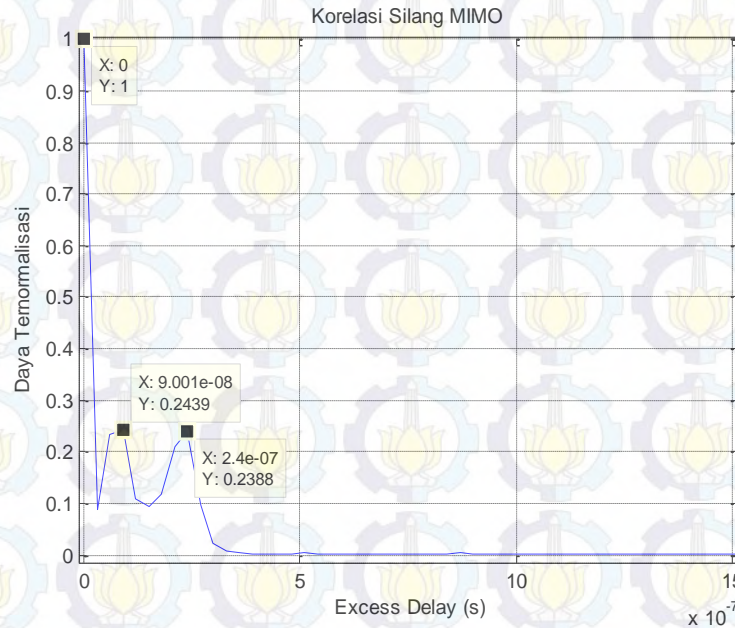
sample impuls, maka impuls sample ke-n tersebut bukan impuls noise. Namun sebaliknya jika impuls sample ke-n tidak bersebelahan dengan impuls sample ke n-1 maupun impuls sample ke n+1, maka impuls sample ke-n tersebut adalah impuls noise.

Setelah dilakukan pengecekan tetangga, langkah berikutnya adalah cek konsistensi munculnya peak dari respon impuls, metodenya adalah sebagai berikut, dari keempat respon impuls tersebut yang berjumlah NK (N menyatakan jumlah respon impuls dan K menyatakan banyak sample dalam respon impuls tersebut) dicek puncak – puncak impuls tersebut yang konsisten kemunculan. Puncak-puncak yang tidak muncul secara konsisten dianggap noise dan dibuang dari himpunan sinyal-sinyal lintasan jamak. Kemunculan suatu komponen lintasan jamak dianggap konsisten jika itu terjadi lebih besar dari atau sama dengan 75% kali jumlah impuls [14] . Apabila impuls tersebut terdeteksi konsisten, maka langkah berikutnya adalah mencari puncak rata-rata impuls tertinggi. Pada Gambar 3.9 merupakan grafik prosentase kemunculan terhadap waktu delay.



Gambar 3.9 Grafik Hasil Pengecekan Konsistensi

Hasil Konsistensi tersebut menunjukkan bahwa waktu delay ke 0 ns – 272 ns bersifat konsisten, nilai estimasi delay yang diinputkan sebagai parameter estimasi delay adalah puncak impuls tertinggi dari hasil crosscorelasi keempat respon impuls tersebut yang ditunjukkan pada Gambar 3.10.



Gambar 3.10 Korelasi Silang MIMO

Estimasi delay $\hat{\tau}'_l$ didapatkan dari proses kroskorelasi sinyal IQ yang diterima dengan data yang dikirim dengan asumsi magnitude terbesar dari hasil croscorelasi tersebut adalah komponen LOS (*line of sight*).

Tahap selanjutnya adalah sinyal IQ dan $\hat{\tau}'_l$ diproses oleh SAGE, untuk mengestimasi sudut ($\widehat{\zeta}_{1,l}$ dan $\widehat{\zeta}_{2,l}$) dan amplitudo kompleks (\hat{a}_l'). Proses ekspektasi dan maksimisasi ditentukan sesuai dengan (2.15) dan (3.5). Sedangkan untuk proses preinisialisasi dilakukan untuk parameter sudut dan amplitudo kompleks saja (diberi nilai sama dengan nol). Langkah berikutnya adalah menginputkan frekuensi yang sesuai dengan pengukuran yang sangat berpengaruh terhadap hasil estimasi, dan penyusunan matrik data pengukuran harus sesuai matrik dari simulasi SAGE.

Simulasi SAGE yang mengestimasi sudut dulu, baru delay, dan amplitudo kompleks tidak bisa diterapkan pada data pengukuran dimana untuk hasil estimasi sudut tidak terdeteksi dengan baik untuk semua komponen sinyal yang terdeteksi. Begitu halnya dengan simulasi SAGE mengestimasi delay, sudut, dan amplitudo kompleks tidak bisa diterapkan pada data pengukuran dimana untuk hasil estimasi delay tidak bisa dipisahkan antar komponen yang terdeteksi, sehingga parameter delay didapatkan dari proses korelasi silang.

BAB 4

HASIL DAN PEMBAHASAN

Hasil penelitian ini diperoleh dari analisa estimasi parameter kanal dari data sintetik untuk antenna array planar dan antenna array linier dengan membandingkan hasil estimasi simulasi sudut dulu dan simulasi delay dulu pada SAGE. Validasi algoritma menggunakan data pengukuran dengan antenna linier, dimana hasil pengukuran divalidasi dengan ray tracing.

Respon Impuls yang didapatkan dari pengukuran ($H(k)$) menggunakan USRP di Hall Gedung Baru PENS sebagai validasi, yaitu komponen LOS dan komponen akibat dua pantulan. Keluaran dari data pengukuran adalah parameter delay, sudut, dan amplitudo kompleks yang akan dibandingkan dengan hasil ray tracing lokasi pengukuran tersebut.

Simulasi algoritma SAGE dalam mengestimasi parameter kanal tidak 100 % akurat, oleh sebab itulah perlu dilakukan menghitung nilai RMSE (*Root Mean Square Error*) pada semua metode simulasi, dimana setiap metode tersebut iterasi 10 kali. Ada empat metode yang digunakan untuk menguji kinerja SAGE, di antaranya adalah kinerja SAGE terhadap perubahan nilai SNR, perubahan jumlah variasi multipath (L), perubahan sudut elevasi, dan kelebihan simulasi sudut dulu dibandingkan simulasi estimasi delay dulu yang diterapkan pada antenna array planar dan antenna array linier.

Setup parameter yang dinisialisasikan sebelum melakukan simulasi adalah menetapkan parameter antenna planar maupun linier seperti jarak antar elemen $\frac{1}{2} \lambda$ pada frekuensi 2.450 GHz (setara 6 cm). Sample deretan pn sequence ditetapkan 510 sample. Parameter sinyal yang perlu dibangkitkan secara acak adalah azimuth dan elevasi dari DOA dan DOD untuk planar dalam rentang $0-2\pi$ dan $0-\pi$, sedangkan untuk linier hanya parameter sudut elevasi dari DOA maupun DOD. Nilai delay dibangkitkan secara acak sepanjang 510 sample. Sedangkan amplitudo kompleks dan nilai noise juga dibangkitkan secara acak. Semua parameter terlihat jelas dibangkitkan acak, sehingga nilai parameter sinyal tersebut dapat diuji.

Kinerja SAGE terhadap pendeteksian parameter delay pada urutan pertama dibandingkan dengan kinerja SAGE yang mendeteksi parameter sudut pada urutan

pertama, dimana untuk memperjelas hasil simulasi tersebut maka label yang digunakan untuk kedua simulasi tersebut adalah :

SAGE 1 : simulasi SAGE dengan mendeteksi parameter delay pada urutan pertama

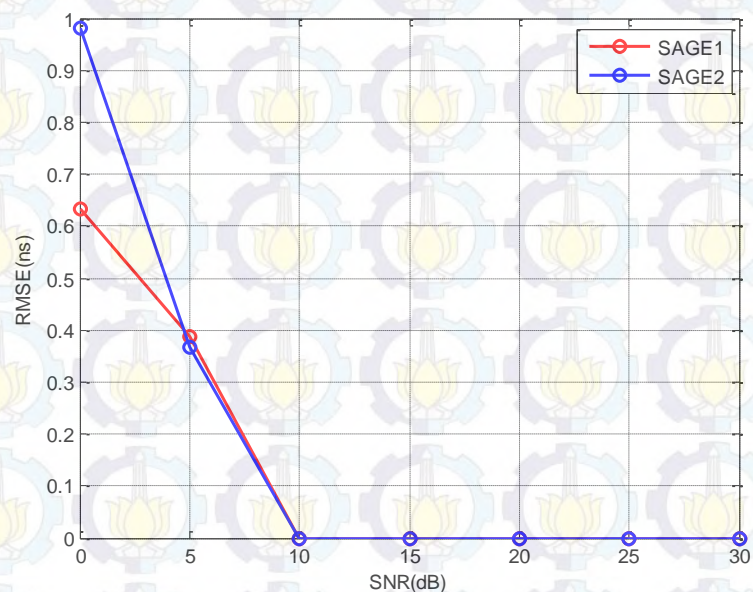
SAGE 2 : simulasi SAGE yang mendeteksi parameter sudut pada urutan pertama.

4.1 Akurasi Estimasi Algoritma SAGE sebagai Fungsi SNR

Analisa akurasi estimasi algoritma SAGE dilakukan pada satu sumber sinyal ($L = 1$) dengan merubah nilai SNR dimulai dari 0 dB sampai dengan 30 dB dengan interval 5 dB. Analisa ini diterapkan pada antenna planar dan linier untuk simulasi SAGE 1 dan simulasi SAGE 2. Faktor adanya multipath dan noise AWGN sangat berpengaruh terhadap karakteristik kanal. Faktor tersebut dapat mengakibatkan sinyal yang diterima mengalami peredaman, pergeseran fase, dan penundaan sinyal yang diterima. Oleh sebab itulah, analisa perubahan SNR terhadap hasil estimasi SAGE berpengaruh.

4.1.1 Perubahan SNR pada Antena Array Planar

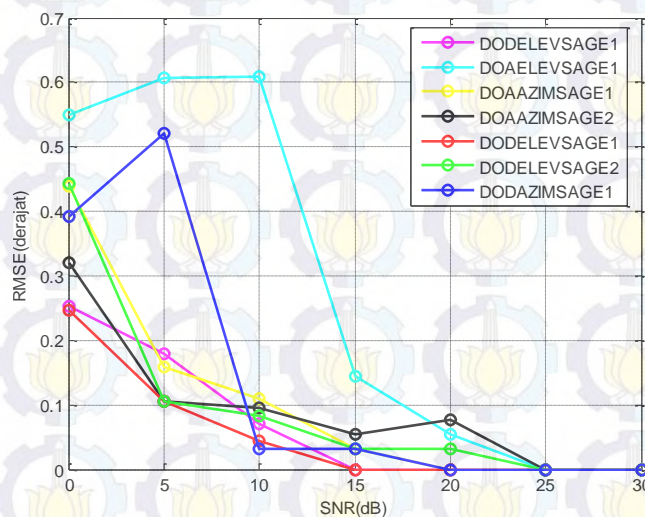
Perubahan nilai SNR dapat mempengaruhi hasil estimasi parameter kanal yaitu delay, sudut, dan amplitudo kompleks. Grafik perubahan nilai SNR disajikan terhadap nilai RMSE dari setiap estimasi parameter kanal.



Gambar 4.1 Kinerja SAGE Terhadap Variasi Nilai SNR untuk Estimasi Parameter Delay pada Antena Planar

Perubahan SNR berpengaruh terhadap estimasi delay, dimana estimasi delay akurat ketika $\text{SNR} > 10 \text{ dB}$, sedangkan hasil estimasi delay untuk simulasi SAGE 2 lebih akurat dibandingkan dengan simulasi SAGE 1, hal ini terlihat ketika SNR 0 dB dan 5 dB. Nilai RMSE delay berdasarkan grafik di atas antara 0.9 – 0.3 ns masih bisa ditoleransi karena nilai rmse masih berada di bawah sample interval delay 1ns. Selain itu pada SNR = 0 dB menunjukkan bahwa simulasi SAGE 2 memberikan RMSE lebih kecil yaitu 0.6317 ns dibandingkan dengan rms eror simulasi SAGE1 yaitu 0.9803 ns. Grafik tersebut membuktikan bahwa kinerja SAGE pada simulasi SAGE2 dapat mengestimasi delay dengan baik

Perubahan SNR pun juga mempengaruhi estimasi sudut (DOA dan DOD) yang ditunjukkan pada Gambar 4.2. Pada grafik tersebut label untuk DOAELEVSAGE1 artinya estimasi sudut elevasi DOA untuk simulasi SAGE1, sedangkan DOAAZIMSAGE1 menyatakan estimasi sudut azimuth untuk simulasi SAGE1. Label DODELEVSAGE1 menyatakan estimasi elevasi DOD untuk simulasi SAGE2. Sedangkan label DODAZIMSAGE2 menyatakan estimasi azimuth DOD untuk simulasi SAGE 2.



Gambar 4.2 Kinerja SAGE Terhadap Variasi Nilai SNR untuk Estimasi Parameter Sudut pada Antena Array Planar

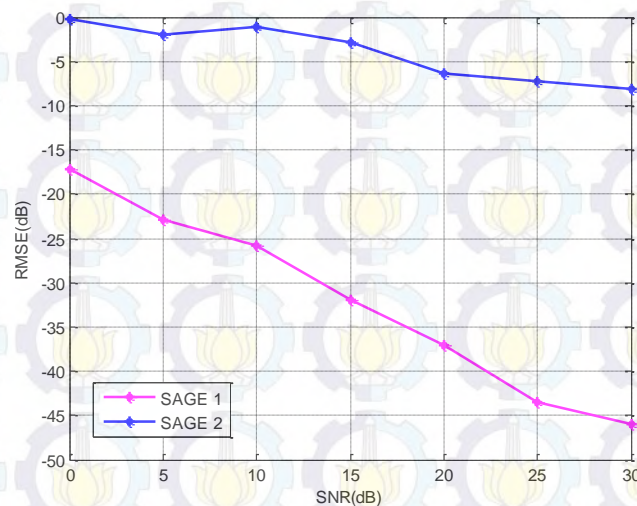
Perubahan nilai SNR berpengaruh juga terhadap estimasi sudut, ketika SNR 0 dB RMSE sudut pada simulasi SAGE2 bernilai lebih kecil dibandingkan dengan

RMSE sudut untuk simulasi SAGE1 (jelas terlihat pada tabel 1). Simulasi SAGE1 pada $\text{SNR} = 0 \text{ dB}$ nilai RMSE estimasi DOA lebih besar 0.3225 derajat dibandingkan nilai RMSE estimasi DOA sebesar 0.253 ns . Selain itu pada $\text{SNR} \geq 20\text{dB}$ RMSE dari estimasi sudut untuk simulasi SAGE2 terdeteksi konvergen, dimana sudut estimasi bernilai sama dengan sudut inputnya atau RMSE bernilai 0° .

Tabel 4.1. RMSE Sudut terhadap perubahan nilai SNR

| snr | SAGE1 | | | | SAGE2 | | | |
|-----|---------|---------|---------|---------|---------|---------|---------|---------|
| | doaelev | doaazim | dodelev | dodazim | doaelev | doaazim | dodelev | dodazim |
| 0 | 0.3225 | 0.5495 | 0.3209 | 0.4427 | 0.253 | 0.4393 | 0.247 | 0.3912 |
| 5 | 0.1449 | 0.6066 | 0.1049 | 0.1049 | 0.1789 | 0.1581 | 0.1049 | 0.5206 |
| 10 | 0.1304 | 0.6099 | 0.0949 | 0.0837 | 0.0707 | 0.1095 | 0.0447 | 0.0316 |
| 15 | 0.1449 | 0.1449 | 0.0548 | 0.0316 | 0 | 0.0316 | 0 | 0.0316 |
| 20 | 0.1049 | 0.0548 | 0.0775 | 0.0316 | 0 | 0 | 0 | 0 |
| 25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Gambar 4.2 dan Tabel 4.1 membuktikan bahwa SAGE dapat mengestimasi sudut doa dan dod dengan baik ketika $\text{SNR} \geq 25 \text{ dB}$, Hasil estimasi sudut pada simulasi SAGE 1 dan SAGE 2 dikatakan akurat karena nilai $\text{RMSE} < 1^\circ$

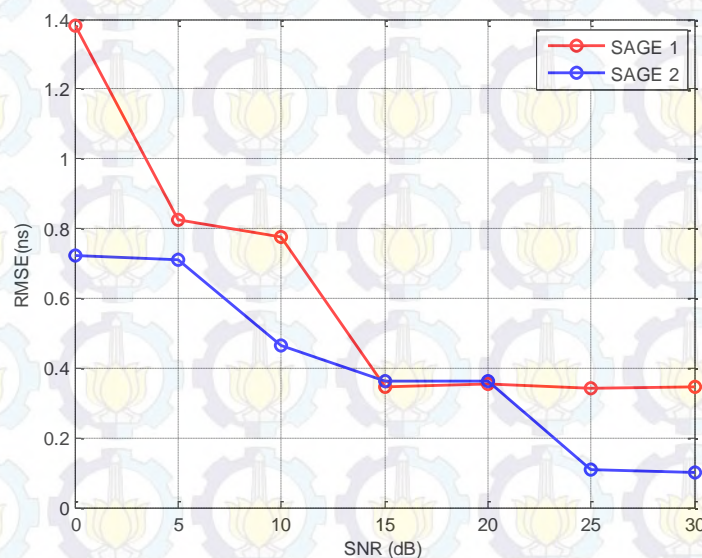


Gambar 4.3 Kinerja SAGE terhadap Variasi Nilai SNR untuk Estimasi Parameter Amplitudo Kompleks pada Antena Array Planar

Perubahan nilai SNR juga mempengaruhi estimasi amplitudo kompleks yang dibangkitkan secara acak. Pada Gambar 4.3 merupakan RMSE amplitudo kompleks antara hasil estimasi dengan inputan amplitudo kompleks yang dibangkitkan, dari gambar tersebut terlihat bahwa hasil estimasi amplitudo kompleks akurat untuk simulasi SAGE2 dimana ketika $\text{SNR} = 0 \text{ dB}$, RMSE tersebut bernilai 0 dB , sedangkan untuk simulasi SAGE1 RMSE bernilai -18 dB . Hal ini dapat disimpulkan bahwa SAGE dapat mengestimasi amplitudo kompleks dengan baik ketika $\text{SNR} = 30 \text{ dB}$ dengan nilai RMSE -9 dB untuk simulasi SAGE2 dan -46 dB untuk simulasi SAGE1.

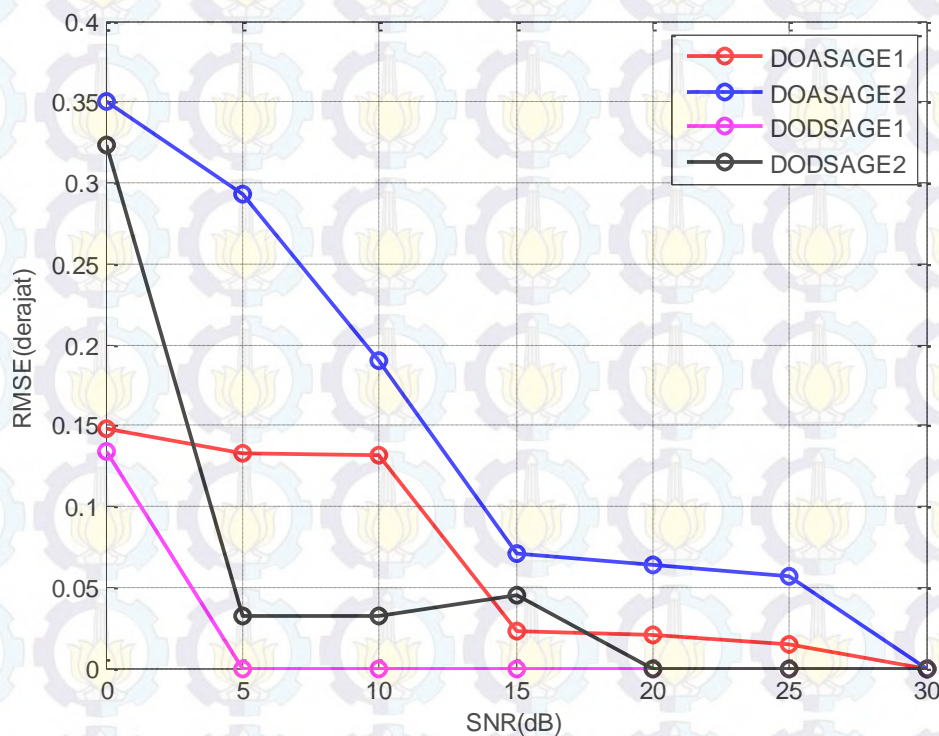
4.1.2 Perubahan SNR pada Antena Array Linier

SAGE bisa diterapkan pada geometri antena sebarang yang merupakan kelebihanannya, akan tetapi jumlah elemen antena mempengaruhi estimasi delay seperti perbandingan Gambar 4.1 dan 4.4 pada $\text{SNR} 0 \text{ dB}$. Untuk antena array planar RMSE estimasi delay antara $0.9 - 0.6 \text{ ns}$, sedangkan untuk antena array linier RMSE estimasi delay terletak antar $0.7 - 1.4 \text{ ns}$. Semakin rapat sample temporal atau semakin banyak jumlah elemen antena, dapat memperkecil kesalahan estimasi delay yang terjadi.



Gambar 4.4 Kinerja SAGE terhadap Variasi nilai SNR terhadap Parameter Delay pada Antena Array Linier

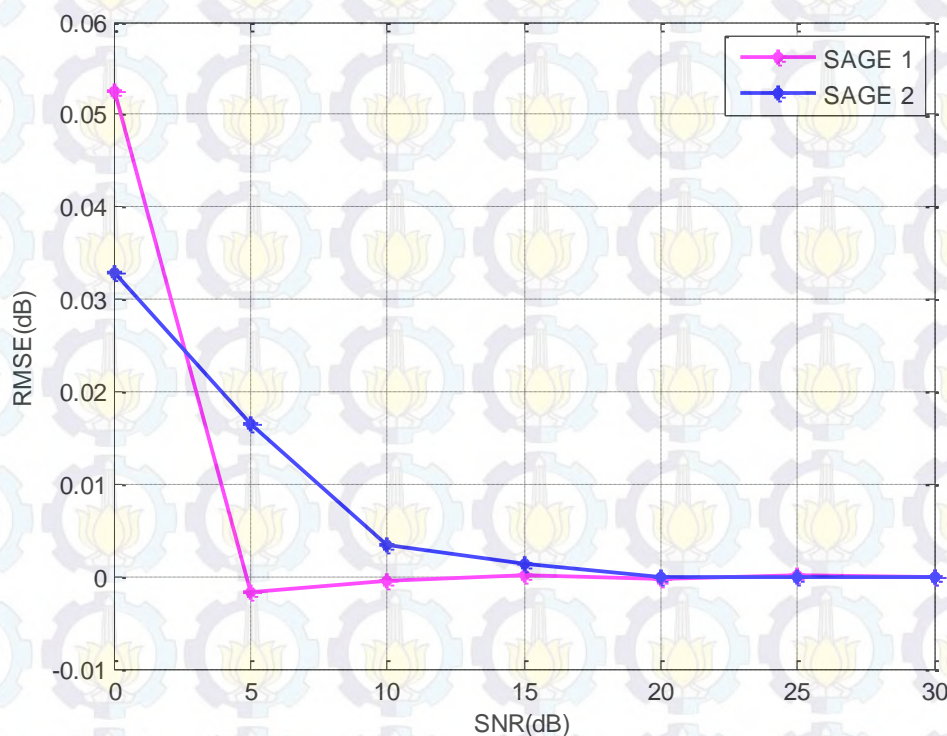
Selain itu, kinerja SAGE dalam simulasi SAGE2 pada antenna linier menunjukkan RMSE estimasi delay lebih kecil dibandingkan dengan RMSE estimasi delay dari hasil simulasi SAGE1. Ketika $\text{SNR} = 30 \text{ dB}$ nilai RMSE dari estimasi delay untuk simulasi SAGE1 dan SAGE2 berturut-turut adalah 0.3464 ns dan 0.1005 ns . Sedangkan ketika $\text{SNR} = 15 \text{ dB}$ dan 20 dB RMSE kedua simulasi ini hampir memiliki RMSE yang hampir sama, akan tetapi tinjauan RMSE untuk kedua simulasi ini adalah ketika $\text{SNR} \geq 20 \text{ dB}$. Selain itu, perubahan nilai snr dapat mempengaruhi estimasi parameter sudut yang ditunjukkan pada Gambar 4.5.



Gambar 4.5 Kinerja SAGE terhadap Variasi nilai SNR terhadap Parameter Sudut pada Antena Array Linier

Estimasi sudut pada antenna linier untuk perubahan SNR membuktikan bahwa semakin besar nilai SNR, estimasi sudut tersebut mendekati benar, seperti ketika nilai $\text{SNR} = 30 \text{ dB}$, nilai estimasi sudut benar dengan nilai RMSE sama dengan nol. Input sudut yang diberikan merupakan sudut elevasi yang acak antara $0-\pi$ dan sudut azimuth $0-2\pi$, yang teriterasi 10 kali, dimana setiap satu kali iterasi sudut input nya berbeda-beda. Hasil RMSE estimasi sudut elevasi DOA dan DOD pada antenna linier lebih kecil dibandingkan rmse estimasi sudut elevasi DOA dan

DOD pada antenna planar ketika SNR 0 dB, hal ini disebabkan karena inputan sudut bersifat acak, sehingga memungkinkan terdapat sudut inputan di luar radiasi planar maupun linier. Selain itu pada grafik tersebut, menunjukkan bahwa SAGE dapat mengestimasi sudut DOD lebih baik dibandingkan sudut DOA yang ditunjukkan pada nilai SNR 5 dB RMSE estimasi sudut DOA pada simulasi SAGE2 adalah 0.1327 ns, sedangkan nilai RMSE DOD adalah 0 ns. Berdasarkan hasil tersebut telah terbukti bahwa estimasi sudut DOD lebih baik dibandingkan sudut DOA pada simulasi SAGE2 maupun simulasi SAGE1.



Gambar 4.6 Kinerja SAGE terhadap Variasi nilai SNR terhadap Parameter Amplitudo Kompleks pada Antena Array Linier

SNR merupakan perbandingan daya sinyal terhadap daya noise, apabila SNR semakin besar hasil estimasi amplitudo akurat. SAGE mampu memisahkan sinyal dengan noise yang ditunjukkan dengan amplitudo kompleks. Hasil estimasi amplitudo kompleks pada Gambar 4.6 menunjukkan bahwa ketika $\text{SNR} = 0 \text{ dB}$, RMSE untuk simulasi SAGE2 0.032dB, sedangkan RMSE untuk simulasi SAGE1 0.052dB, Hasil tersebut dapat disimpulkan bahwa kinerja algoritma SAGE untuk

simulasi SAGE2 lebih baik. Pada $\text{SNR} \geq 15 \text{ dB}$, Rasio dari amplitudo estimasi dengan amplitudo kompleks yang dibangkitkan bernilai 0 dB, hal ini menunjukkan estimasi amplitudo kompleks dengan amplitudo input bernilai sama (akurat). Hasil estimasi amplitudo kompleks tersebut dapat disimpulkan juga bahwa kinerja SAGE dapat mengestimasi amplitudo kompleks pada SNR 15 dB.

Dari Grafik perubahan SNR di atas dapat disimpulkan bahwa SNR minimal yang dibutuhkan untuk algoritma SAGE untuk mengestimasi parameter kanal dengan baik pada antenna array planar adalah 10 dB, sedangkan untuk antenna linier membutuhkan minimal nilai SNR 30 dB.

4.2 Akurasi Estimasi Algoritma SAGE sebagai Fungsi Jumlah Multipath Sinyal

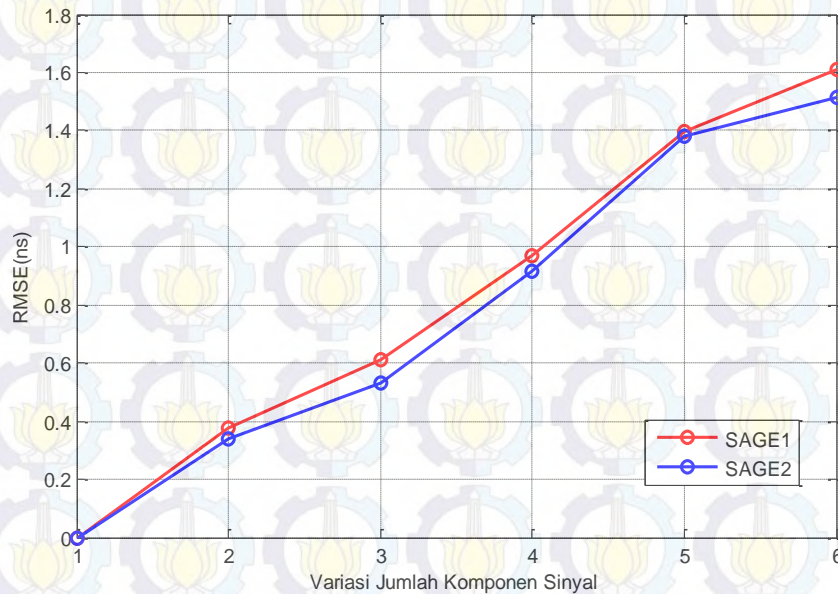
Ketika sinyal gelombang elektromagnetik dipancarkan, maka sinyal akan merambat melalui media udara yang mengalami mekanisme propagasi sebelum sampai ke penerima. Mekanisme tersebut berupa pantulan, *scattering*, dan difraksi yang mengakibatkan sinyal melewati lintasan multipath sebelum sampai di penerima. *Multipath* sangat mempengaruhi karakteristik kanal, terutama untuk sinyal datang pada antenna lebih dari satu sinyal, sehingga sinyal yang sampai di penerima adalah merupakan replikasi sinyal yang sama, namun dengan fase, amplitudo dan delay waktu yang berbeda.

Semakin banyak sinyal yang diterima oleh penerima menyebabkan noise yang diterimapun semakin besar, sehingga membuat proses dekomposisi sinyal pada E-Step menjadi lebih kompleks, hal ini membuat hasil estimasi SAGE kurang akurat, sehingga pada subbab ini dilakukan pengamatan perubahan nilai *multipath* terhadap kinerja SAGE pada antenna array planar maupun antenna linier.

4.2.1 Perubahan Jumlah Multipath Sinyal Pada Antena Array Planar

Perubahan variasi jumlah *multipath* juga mempengaruhi kinerja SAGE pada antenna array planar dalam mengestimasi delay, DOA, DOD, dan amplitudo kompleks. Variasi jumlah sinyal dilakukan mulai dari 1 – 6 lintasan multipath, selain itu jumlah elemen antenna dalam karakteristik kanal akibat adanya multipath

sangat mempengaruhi kinerja SAGE. Semakin besar jumlah elemen antenna yang digunakan semakin besar pula sampling temporal untuk mendeteksi lintasan sinyal dengan baik, oleh sebab itulah pada penelitian ini pengamatan perubahan jumlah lintasan *multipath* dibatasi sampai 6 lintasan baik diterapkan pada antenna planar maupun antenna linier.

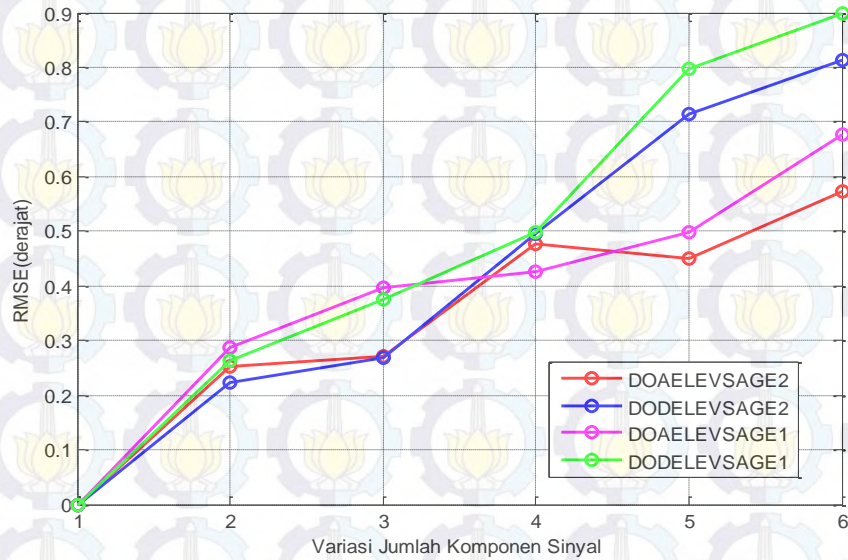


Gambar 4.7 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Delay pada Antena Array Planar

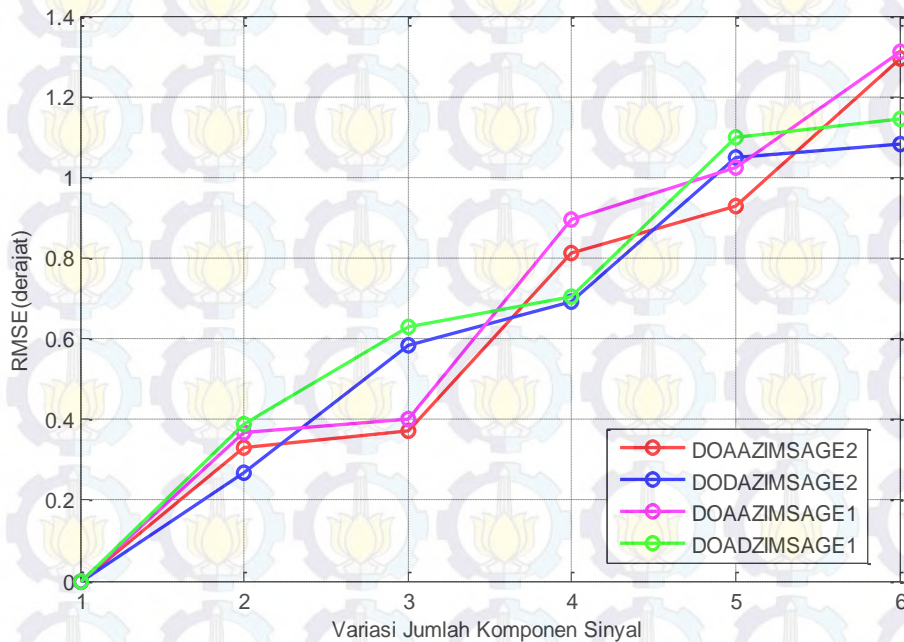
Jumlah Komponen sinyal mempengaruhi estimasi delay baik untuk simulasi SAGE2 maupun simulasi SAGE1 yang ditunjukkan pada Gambar 4.7. Saat $L = 6$ estimasi delay mengalami kesalahan sebesar 0.573 ns (simulasi SAGE2) dan 0.676 ns (simulasi SAGE1).

Selain itu estimasi sudut juga dipengaruhi oleh jumlah komponen sinyal, pada Gambar 4.8 membuktikan bahwa estimasi elevasi untuk arah keberangkatan dan kedatangan sinyal mengalami kesalahan untuk jumlah komponen yang lebih besar. Hal ini menunjukkan bahwa RMSE relative besar untuk $L = 6$, sehingga kinerja SAGE untuk mengestimasi sudut dengan jumlah komponen sinyal ($L >$ elemen antenna) kurang akurat. Kesalahan tersebut disebabkan oleh nilai beda sudut elevasi yang besar antar komponen sinyal yang datang di penerima, sehingga

menyebabkan hasil estimasi untuk sudut lebih besar, akan tetapi nilai RMSE yang didapatkan relative kecil karena $< 1^\circ$.



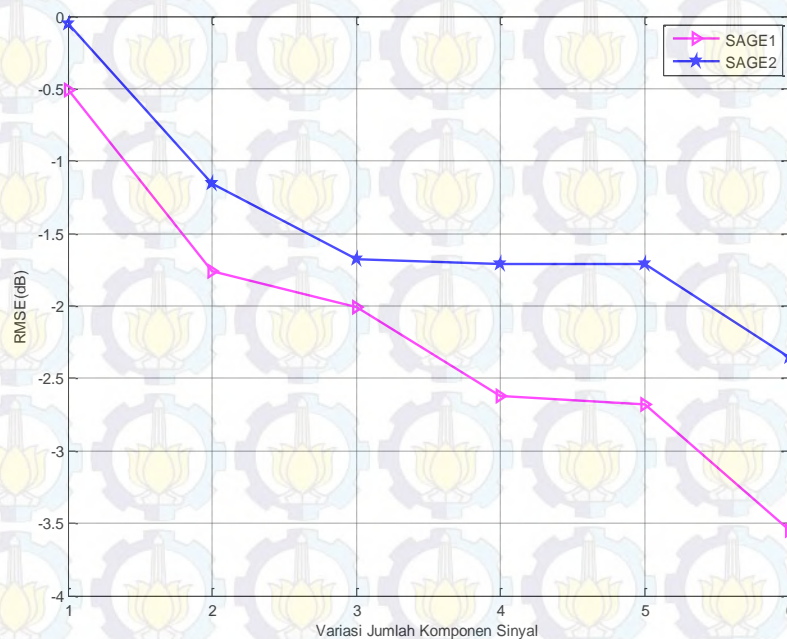
Gambar 4.8 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut Elevasi (DOA dan DOD) pada Antena Array Planar



Gambar 4.9 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut Azimuth (DOA dan DOD) pada Antena Array Planar

Selanjutnya, pengaruh estimasi sudut azimuth (DOA dan DOD) terhadap perubahan SNR yang ditunjukkan pada Gambar 4.9, Nilai RMSE yang didapatkan besar ketika $L = 6$ yaitu untuk simulasi SAGE2 RMSE azimuth DOA 1.29° dan RMSE azimuth DOD 1.08° , sedangkan simulasi SAGE1 RMSE azimuth DOA 1.3° dan RMSE azimuth DOD 1.14° . Berdasarkan Gambar 4.8 dan 4.9 membuktikan bahwa estimasi sudut elevasi lebih baik dibandingkan dengan estimasi sudut azimuth ketika $L = 5$.

Hasil analisa untuk perubahan jumlah komponen sinyal maupun perubahan SNR membuktikan bahwa kinerja SAGE pada simulasi SAGE2 lebih baik dibandingkan dengan simulasi SAGE1. Akurasi SAGE pada tahap ini (perubahan jumlah komponen sinyal untuk estimasi sudut) dapat disimpulkan bahwa SAGE mampu mengestimasi parameter sinyal yang datang dari beberapa sinyal akibat lintasan *multipath*.

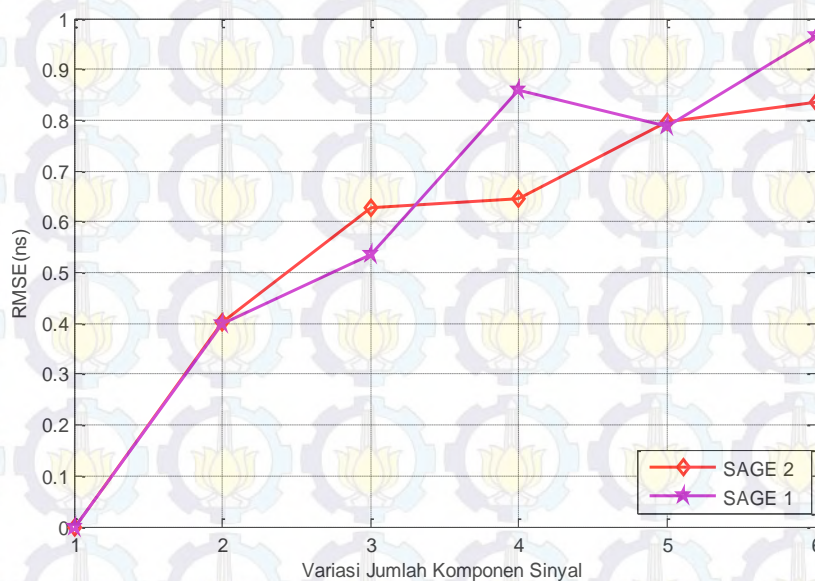


Gambar 4.10 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Amplitudo Kompleks (DOA dan DOD) pada Antena Array Planar

Parameter kanal selanjutnya yang terestimasi adalah amplitudo kompleks ditunjukkan pada Gambar 4.10. Estimasi tersebut dilakukan dengan membagi hasil estimasi amplitudo kompleks dengan hasil amplitudo kompleks input. Pada grafik tersebut terlihat bahwa redaman semakin besar dengan meningkatnya jumlah komponen sinyal dalam rentang nilai -0.5 dB sampai -3.5 dB (hasil SAGE1), hasil estimasi amplitudo kompleks untuk simulasi SAGE2 adalah -0.056 dB sampai -2.35 dB, Hasil ini membuktikan bahwa algoritma SAGE dapat melakukan estimasi amplitudo kompleks dengan akurat.

4.2.2 Perubahan Jumlah Multipath Sinyal Pada Antena Array Linier

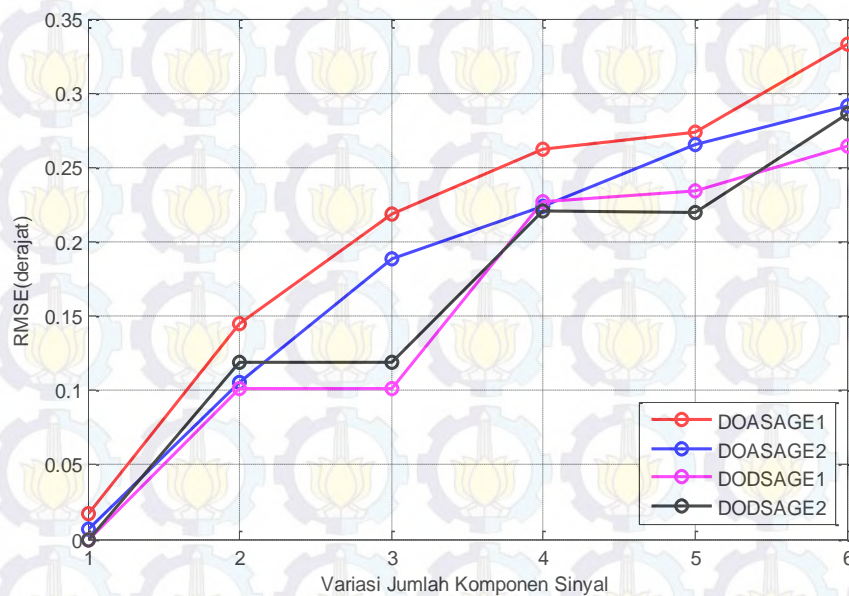
Perubahan jumlah sinyal multipath juga mempengaruhi hasil dari estimasi delay, sudut, dan amplitudo kompleks pada antena linier yang ditunjukkan pada Gambar 4.11 sampai 4.12. Dari hasil tersebut dapat dibuktikan bahwa estimasi delay pada antena linier juga dipengaruhi oleh jumlah komponen sinyal datang.



Gambar 4.11 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Delay pada Antena Array Linier

Estimasi delay untuk simulasi SAGE2 dan simulasi SAGE1 ketika $L = 5$, memiliki nilai RMSE delay hampir sama yaitu 0.79 ns dan 0.78 ns. Ketika $L = 3$ estimasi delay untuk simulasi SAGE1 memiliki RMSE lebih kecil dibandingkan dengan

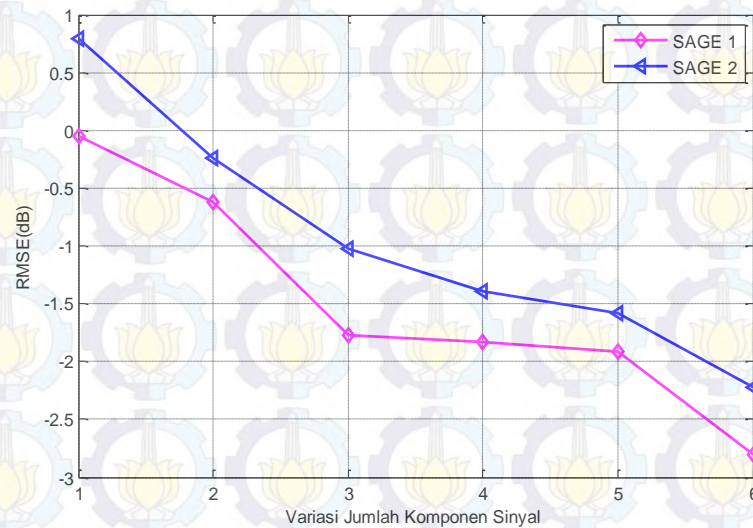
simulasi SAGE2 yaitu 0.535ns dan 0.626, berbeda halnya ketika $L = 4$, keadaanya kebalikan dari $L = 3$ dimana estimasi delay untuk simulasi SAGE1 mempunyai RMSE yang lebih besar dibandingkan RMSE delay simulasi SAGE2. Hal ini terjadi dikarenakan delay yang diinputkan acak setiap iterasi 10 kali, sehingga memungkinkan di antara iterasi tersebut terdapat eror yang besar sehingga hasil simulasi tersebut tidak stabil antara $L = 3, 4$, dan 5 . Meskipun hasil estimasi delay antara simulasi SAGE1 dan simulasi SAGE2 tidak stabil, akan tetapi keduanya mempunyai nilai RMSE di bawah 1 ns, sehingga dapat dikatakan bahwa kinerja SAGE dapat mengestimasi parameter sinyal sampai 6 lintasan dengan akurat .



Gambar 4.12 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Sudut (DOA dan DOD) pada Antena Array Linier

Variasi jumlah lintasan mempengaruhi estimasi sudut pada array linier, berdasarkan Gambar 4.12 nilai RMSE terletak sekitar 0.05° - 0.35° . Ketika $L = 2$, nilai RMSE untuk estimasi sudut DOD dan sudut DOA (simulasi SAGE2) serta sudut DOD (simulasi SAGE1) hampir sama atau berhimpit sedangkan nilai RMSE estimasi DOA untuk simulasi SAGE1 tidak jauh beda dari ketiga nilai RMSE yang berhimpit tersebut, hal ini dikarenakan ketika jumlah multipath sama dengan jumlah elemen antena kemungkinan estimasi mendekati benar baik untuk simulasi SAGE2 maupun SAGE1. Sedangkan ketika $L = 6$, menunjukkan bahwa estimasi

sudut DOD lebih baik dibandingkan dengan estimasi sudut DOA, untuk simulasi SAGE2 didapatkan nilai RMSE DOD 0.2862° dan nilai RMSE DOA 0.291° sedangkan untuk simulasi SAGE1 nilai RMSE DOD 0.2645° dan nilai RMSE DOA 3.326° . Nilai RMSE dari semua grafik pada Bab 4 disediakan pada lampiran.



Gambar 4.13 Kinerja SAGE Terhadap Variasi Jumlah Sinyal untuk Estimasi Amplitudo Kompleks pada Antena Array Linier

SAGE dapat mengestimasi amplitudo kompleks dengan baik, grafik tersebut merupakan rasio perbandingan antara amplitudo kompleks inputan dengan amplitudo kompleks hasil estimasi. Untuk simulasi SAGE2 nilai RMSE didapatkan -0.04 dB sampai -2.8 dB, sedangkan simulasi SAGE1 nilai RMSE didapatkan 0.79 dB sampai -2.22 dB. Grafik tersebut juga menyatakan bahwa redaman semakin besar dengan meningkatnya jumlah *multipath* sinyal. Hasil Kinerja SAGE untuk mengestimasi parameter delay, sudut, dan amplitudo kompleks terhadap perubahan lintasan *multipath* dinyatakan akurat dengan nilai RMSE yang relatif kecil.

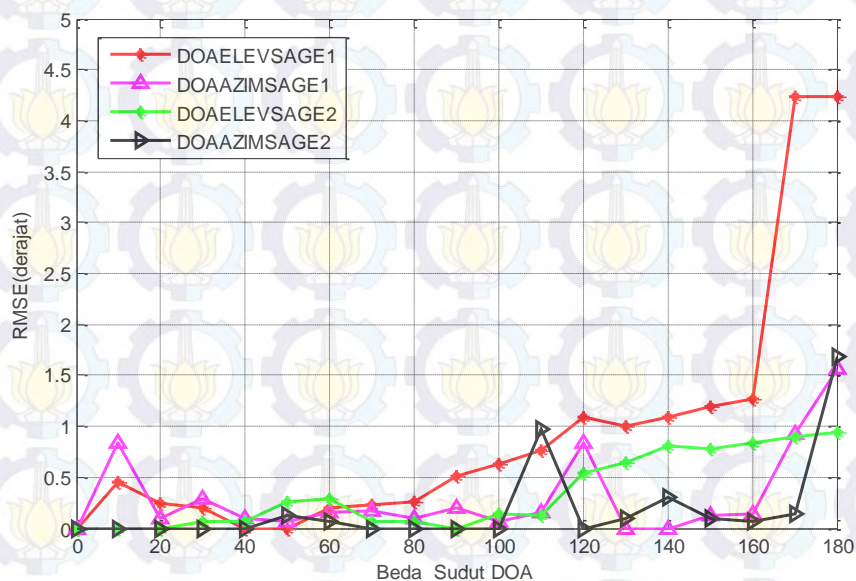
4.3 Akurasi Estimasi Algoritma SAGE sebagai Fungsi Beda Sudut Elevasi

Pada subbab ini akan dibahas kinerja algoritma SAGE terhadap sudut elevasi yang berkaitan dengan pengaruh geometri dari antena planar maupun linier. Sudut elevasi sebagai sudut pengamatan dari DOA maupun DOD.

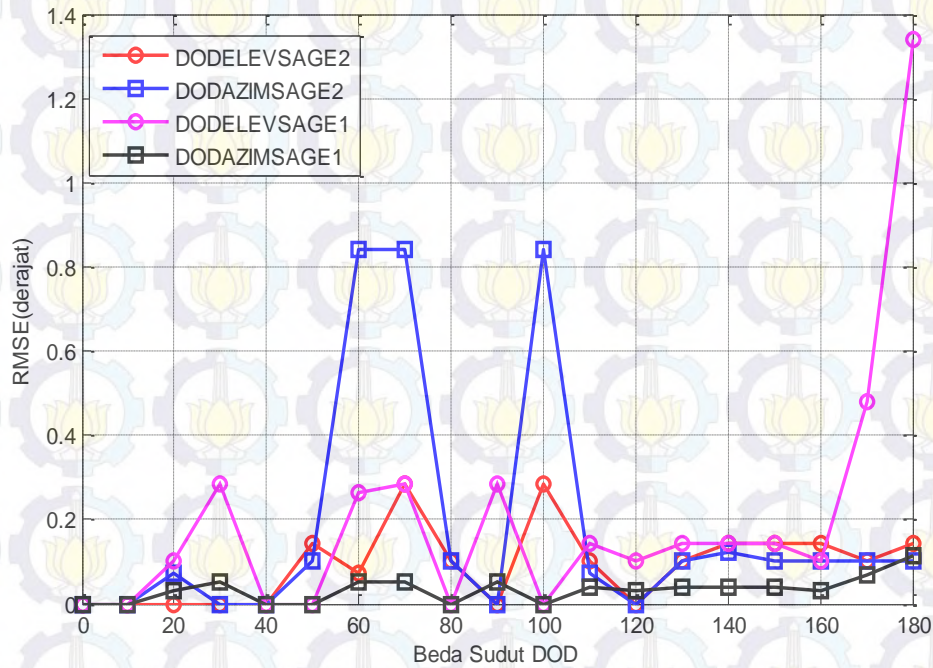
4.3.1 Perubahan Sudut Elevasi Pada Antena Array Planar

Perubahan sudut elevasi mempengaruhi kinerja SAGE dalam mengestimasi sudut. Pada Array planar arah kedatangan dan arah keberangkatan terdiri dari elevasi dan azimuth. Perubahan nilai sudut elevasi dengan nilai sudut azimuth tetap mempengaruhi kinerja SAGE, ketika elevasi elevasi 0° dan 180° nilai estimasi sudut azimuth mengalami kesalahan atau terdeteksi 360° untuk kedua nilai elevasi tersebut.

Analisa lebih lanjut adalah berkaitan dengan beda sudut DOA dan DOD pada antena array planar, dimana terdapat dua komponen lintasan dengan beda sudut DOA yang dimulai dari 0° sampai 180° yang dibangkitkan secara acak memungkinkan semua rentang sudut untuk azimuth $[0, 2\pi]$ dan elevasi $[0, \pi]$ dapat diuji. Gambar 4. 14 membuktikan bahwa kinerja SAGE mampu mengestimasi beda sudut elevasi yang besar. Sudut yang berdekatan atau sudut yang berjauhan mampu diestimasi oleh SAGE, dimisalkan untuk 2 komponen yang mempunyai nilai sudut datang yang sama (beda sudut sama dengan 0°) dapat terdeteksi dengan baik oleh SAGE dibuktikan dengan nilai RMSE nya bernilai nol. Selain itu, SAGE mampu mendeteksi sudut yang berjauhan untuk beda sudut $20^\circ - 160^\circ$ dengan nilai RMSE yang sangat kecil yaitu $< 1^\circ$.



Gambar 4.14 Kinerja SAGE Terhadap Beda Sudut DOA pada Antena Array Planar



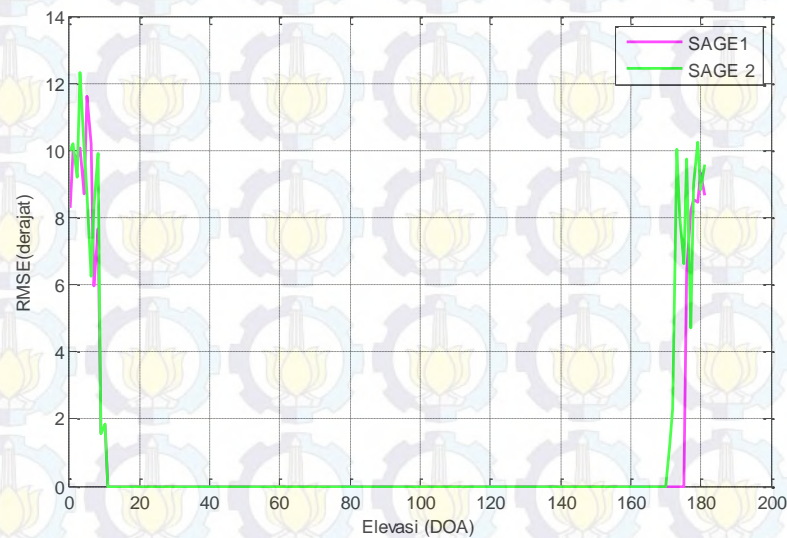
Gambar 4.15 Kinerja SAGE Terhadap Beda Sudut DOD pada Antena Array Planar

Beda sudut DOD yang dibangkitkan secara acak untuk 2 lintasan komponen dapat terdeteksi oleh SAGE dengan rentang 10° yang dimulai dengan beda sudut $0^\circ - 180^\circ$. Kinerja SAGE pada Gambar 4.15 menunjukkan variasi error yang kecil ($< 1^\circ$) untuk sudut elevasi dan azimuth dari arah keberangkatan lintasaan. Variasi error ini pula membuktikan bahwa SAGE dapat mendeteksi beda sudut DOD lebih baik dibandingkan dengan beda sudut DOA.

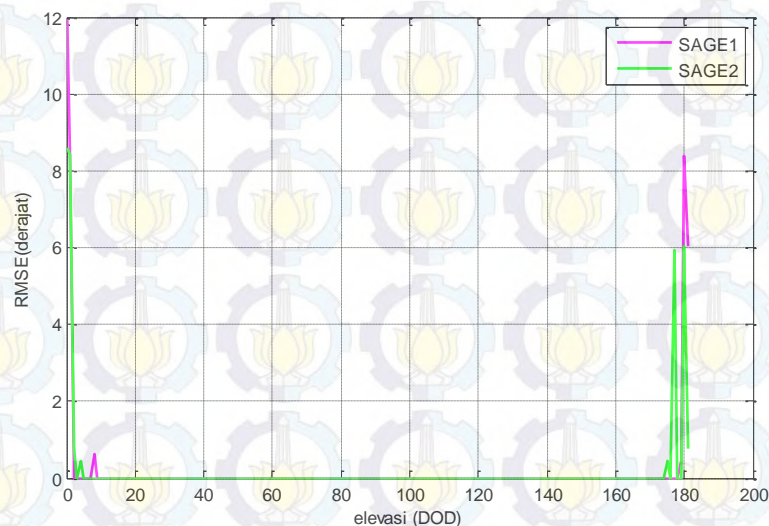
Variasi error beda sudut DOA dan DOD kecil terjadi disebabkan oleh proses dekomposisi sinyal tidak hanya memisahkan data lengkap antar lintasan yang berdekatan namun juga memisahkan data lengkap dengan noise, pada kedua gambar tersebut terlihat bahwa SAGE dapat melakukan estimasi sudut cukup akurat untuk semua beda sudut kedatangan dan keberangkatan dengan nilai error $< 1^\circ$. Akan tetapi ketika beda sudut tersebut sampai mencapai 180° , error yang terjadi cukup besar dikarenakan pengarahannya dari geometri array planar untuk sudut 180° mengalami pelemahan gain.

4.3.2 Perubahan Sudut Elevasi Pada Antena Array Linier

Pada subbab ini akan dibahas pengaruh perubahan sudut elevasi DOA dan DOD pada array linier yang berasal dari satu sumber atau satu sinyal, hal ini berhubungan dengan pola radiasi dari array linier, dimana sudut yang mendekati 0° dan 180° akan mengalami pelemahan gain yang bisa mempengaruhi estimasi sudut elevasi dari DOA maupun DOD yang ditunjukkan pada Gambar 4.16 dan 4.17.



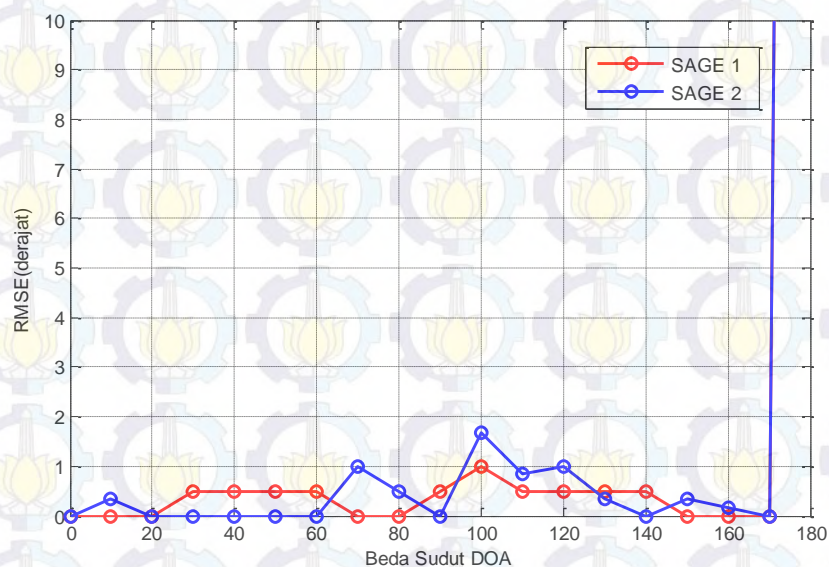
Gambar 4.16 Kinerja SAGE Terhadap Perubahan Sudut Elevasi DOA pada Antena Array Linier



Gambar 4.17 Kinerja SAGE Terhadap Perubahan Sudut Elevasi DOD pada Antena Linier

Gambar 4.16 dan 4.17 tersebut menyatakan bahwa elevasi $0-10^\circ$ (baik DOA maupun DOD) tidak bisa diestimasi dengan baik dimana RMSE tertinggi ketika elevasi DOA pada posisi 5° dan 179° , untuk nilai RMSE nya berturut-turut adalah 11.6° dan 9.3° . Sedangkan elevasi DOD memiliki RMSE nilai tertinggi pada posisi 0° dan 179° , dengan nilai RMSE berturut-turut adalah 8.6° dan 6.3° . Hal ini terjadi karena pola radiasi pada antena array linier mengalami pelemahan gain pada sudut $0^\circ-10^\circ$ dan $170^\circ-180^\circ$, Hal ini membuat SAGE tidak bisa mendeteksi arah kedatangan dan keberangkatan sinyal pada sudut itu baik untuk simulasi SAGE2 maupun simulasi SAGE1.

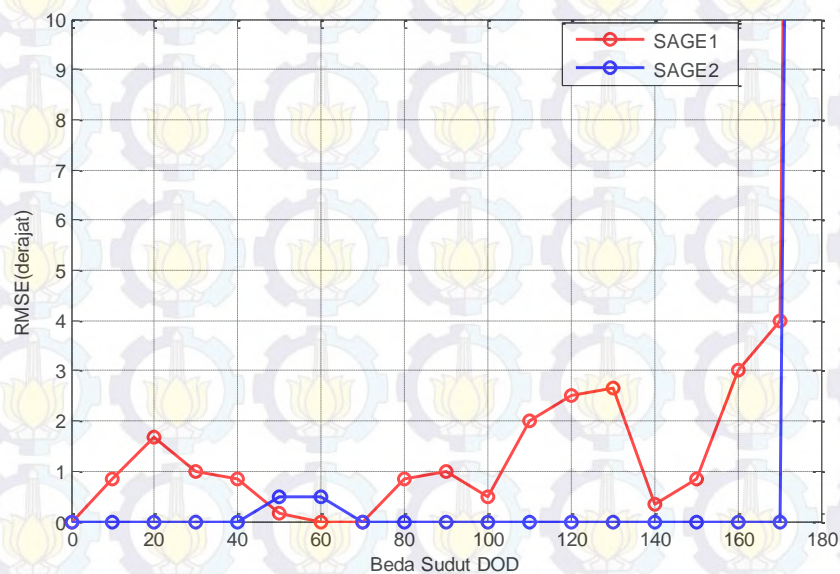
Setelah mengetahui karakteristik kinerja SAGE dari perubahan elevasi, berikutnya juga akan dibahas kinerja SAGE untuk beda sudut elevasi DOA dan DOD dengan rentang yang besar, artinya akan dibangkitkan dua sinyal dengan resolusi sudut kedatangan dan resolusi sudut keberangkatan besar, hasil tersebut ditunjukkan pada Gambar 4.18 dan Gambar 4.19.



Gambar 4.18 Kinerja SAGE Terhadap Bada Sudut DOA pada Antena Array Linier

Pada Gambar 4.18 membuktikan bahwa untuk simulasi SAGE1, beda sudut keberangkatan mempengaruhi sudut estimasi SAGE dimana semakin besar beda sudut keberangkatannya juga RMSE yang dihasilkan juga besar. Akan tetapi untuk

simulasi SAGE2 dengan resolusi sudut besar tidak mempengaruhi hasil estimasi sudut. Sedangkan pada simulasi SAGE1 terlihat bahwa beda sudut keberangkatan yang besar mempengaruhi hasil estimasi sudut, dimana nilai RMSE yang dihasilkan semakin besar. Ketika beda sudut keberangkatan 60° , 100° dan 140° , nilai RMSE yang didapatkan 0° , 0.5° dan 0.333° . sedangkan untuk beda sudut keberangkatan nilai RMSE yang dihasilkan bervariasi. Nilai variasi RMSE yang kecil dengan beda sudut keberangkatan yang besar disebabkan oleh SAGE mampu memisahkan sinyal atau data lengkap dengan noise ketika proses dekomposisi sinyal.



Gambar 4.19 Kinerja SAGE Terhadap Beda Sudut DOD pada Antena Array Linier

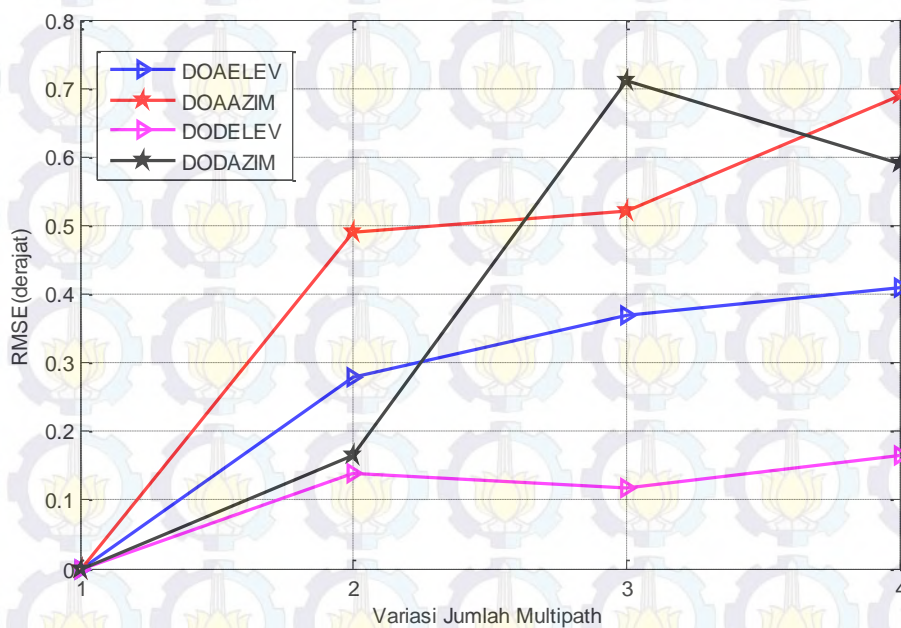
Selanjutnya pengaruh beda sudut kedatangan pada Gambar 4.19 membuktikan bahwa SAGE mampu mendeteksi 2 sinyal datang dengan sudut datang yang besar baik simulasi SAGE1 dan simulasi SAGE2. Ketika 2 sinyal datang dengan beda sudut keberangkatan dan beda sudut kedatangan 180° , SAGE tidak bisa mendeteksi sinyal tersebut, karena terjadi pelemahan gain pada sudut 180° . Sedangkan untuk beda sudut keberangkatan 0° , SAGE mampu mendeteksi 2 sinyal yang datang pada arah yang sama yang ditunjukkan dengan nilai RMSE yang dihasilkan 0° .

Analisa perubahan sudut elevasi terhadap kinerja SAGE telah dibuktikan bahwa estimasi sudut DOA maupun DOD dipengaruhi oleh pola radiasi antena

array linier, dimana untuk sudut 0° dan 180° mengalami pelemahan gain dari radiasi antenna linier sehingga menyebabkan keambiguan dalam mengestimasi sudut.

4.4 Kelebihan Kinerja SAGE dalam Mengestimasi Parameter Sudut Dulu

Pada sub bab ini akan dijelaskan tentang kelebihan estimasi sudut jika pendeteksian sinyal dimulai dari parameter sudut. Kelebihan dari simulasi SAGE2 yang telah dibuktikan pada sub bab sebelumnya bahwa simulasi SAGE2 memberikan hasil estimasi yang lebih akurat dibandingkan dengan simulasi SAGE1. Selain itu, pada bagian ini akan membuktikan bahwa secara simulasi SAGE2 mendeteksi komponen sinyal yang datang secara bersamaan atau komponen yang terdelay sama.



Gambar 4.20 Kinerja SAGE Mengestimasi Sudut Terhadap Variasi Komponen dengan Delay yang Sama pada Antena Array Planar

Pada Gambar 4.20 menunjukkan hasil estimasi sudut dengan delay yang sama pada antenna planar. Hasil tersebut menyatakan bahwa estimasi sudut masih dideteksi dengan baik meskipun dengan sinyal input yang terdelay sama dimana nilai RMSE yang didapatkan $< 1^\circ$. Ketika hasil estimasi sudut DOA elevasi, DOA

azimuth, DOD elevasi, dan DOD azimuth ketika $L = 4$, nilai eror yang didapatkan berturut-turut adalah 0.59° , 0.69° , 0.41° dan 0.165° .

Sedangkan pada Gambar 4.21 menunjukkan kelebihan simulasi SAGE2 pada antenna array linier, dimana sudut tetap terdeteksi dengan baik meskipun sinyal yang datang bersamaan (terdelay sama), dengan nilai RMSE yang terhitung $< 1^\circ$.



Gambar 4.21 Kinerja SAGE Mengestimasi Sudut terhadap Variasi Komponen dengan Delay yang Sama Pada Antena Array Linier

1.5 Validasi Algoritma SAGE berdasarkan Pengukuran di Hall PENS

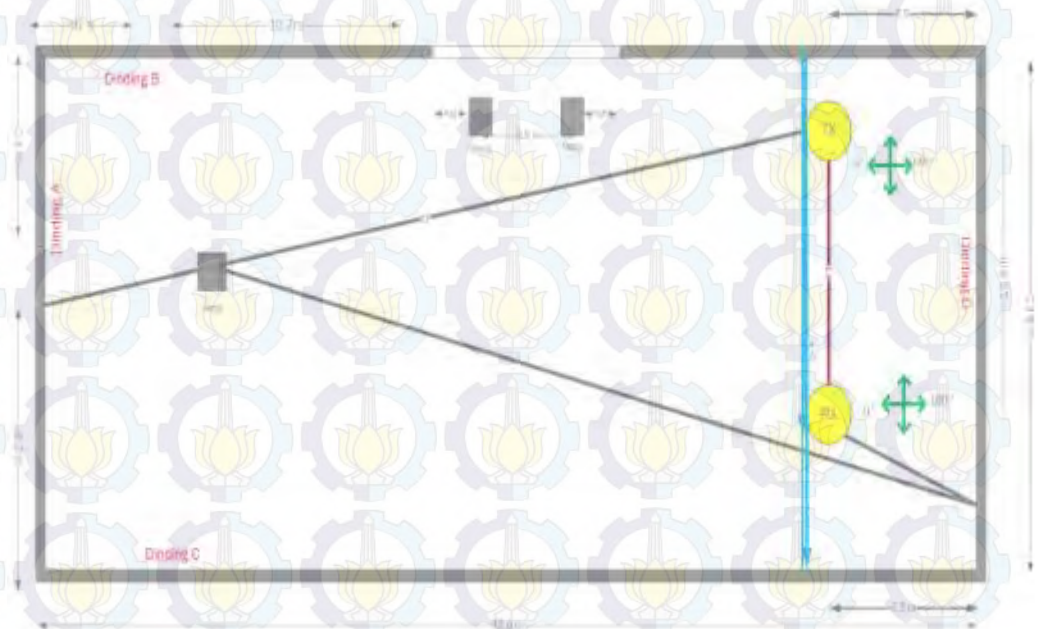
Verifikasi kinerja SAGE dilakukan dengan pengukuran respon kanal di Hall Gedung Baru PENS di frekuensi 2.45 MHz. Kinerja SAGE yang mendeteksi sudut pada urutan pertama tidak bisa diterapkan pada hasil pengukuran, dikarenakan tidak bisa mendeteksi sudut dengan baik dan dibutuhkan teknik preprocessing data pengukuran apabila pendeteksian parameter kanal dimulai dari sudut dulu. Pada subbab ini dijelaskan tentang verifikasi kinerja SAGE dalam pendeteksian delay dulu, dimana estimasi delay tersebut didapatkan dari hasil crosscorelasi data yang diterima dan data yang dikirim.

Respon kanal ($H_{N \times K}$) yang didapatkan dari pengukuran merupakan data kompleks (IQ), sehingga perlu dilakukan korelasi silang antara data yang dikirim dan data yang diterima untuk mendapatkan respon impuls dari pengukuran. Hasil korelasi silang puncak tertinggi diasumsikan sebagai

komponen LOS dan digeser ke titik nol. Proses pengambilan data dan pengolahan data pengukuran telah dijelaskan pada Bab 3 sebelumnya. Data pengukuran yang didapatkan berupa data IQ atau bentuk kompleks dengan dimensi $N \times K$. Penelitian ini dikerjakan dalam satu team peneliti tentang penghambur lokal yang dilakukan oleh Martianda Erste Anggraeni, sehingga data pengukuran yang diolah sama.

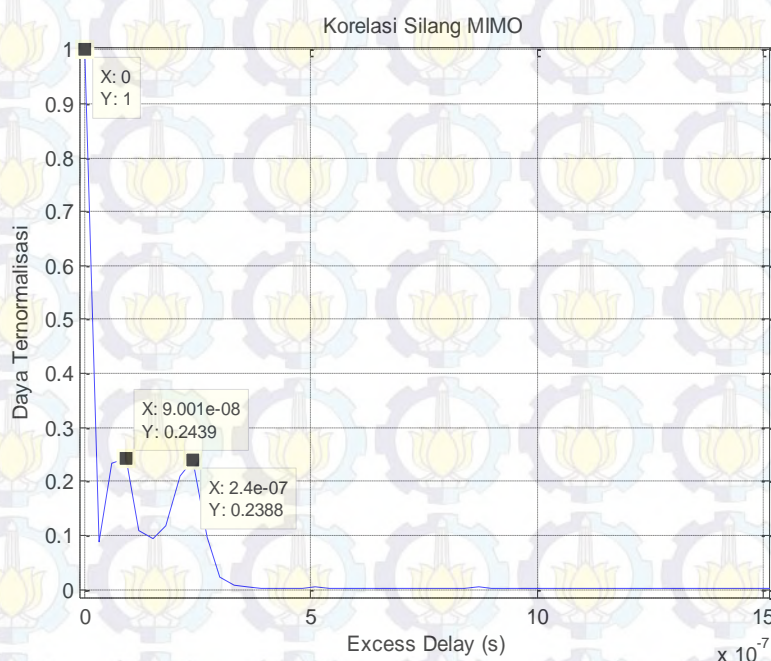
Lokasi pengukuran merupakan ruangan indoor berukuran persegi panjang dengan penghalang tembok, langit-langit (atap gedung), dan beberapa tiang tembok dengan setup pengukuran seperti Gambar 3.6. Pantulan Multipath terjadi karena penghalang yang ada di sekitar lokasi pengukuran, seperti Refleksi terjadi ketika luas penghalang lebih besar dari panjang gelombang, difraksi terjadi di tepi/pinggir dari penghalang yang menyebabkan terjadinya *shadowing*.

Prosedur validasi dari pengukuran dilakukan dengan pencerminan posisi TX dan RX pada ruangan pengukuran, dimana tata ruangan dibuat sama sesuai skala aslinya yang mewakili ruangan, posisi TX dan RX sesungguhnya ditunjukkan pada Gambar 4.22.



Gambar 4.22 Ray Tracing dari Lintasan yang Terdeteksi oleh SAGE

Jumlah lintasan sinyal yang terdeteksi berdasarkan ray tracing lokasi pengukuran sekitar 13 lintasan (13 lintasan yang terdeteksi bisa dilihat secara rinci di tabel lampiran ray tracing) . Akan tetapi, karena keterbatasan peralatan membuat dari 13 lintasan tersebut, hanya 3 lintasan yang terdeteksi dengan baik oleh SAGE. Tiga lintasan tersebut ditunjukkan pada Gambar 4.23 di antaranya adalah lintasan #1 yang merupakan lintasan LOS, lintasan #2 merupakan lintasan yang didapatkan dari pantulan dinding C ke dinding A, dan lintasan #3 merupakan lintasan sinyal yang didapatkan dari pantulan tiang ke dinding B. Ketiga lintasan tersebut merupakan komponen sinyal yang mempunyai daya tertinggi yang terlihat pada respon impuls pada Gambar 4.23 di bawah ini.



Gambar 4.23 Respon Impuls Hasil Pengukuran

Komponen LOS yang terdeteksi mempunyai daya tertinggi dengan delay 0 ns, Komponen ke dua (pantulan dinding C ke dinding A) terdeteksi dengan delay 9 ns, dan komponen ke tiga (pantulan dari tiang ke dinding B) terdeteksi delay 242 ns. Parameter delay tersebut merupakan estimasi delay yang dijadikan input delay pada SAGE untuk mendapatkan parameter kanal yang lain, seperti DOA, DOD dan amplitudo kompleks. Hasil estimasi dari SAGE dan Hasil Ray Tracing ditunjukkan pada tabel 4.2.

Tabel 4.2 Perbandingan Hasil Estimasi Algoritma SAGE berdasarkan Data Pengukuran dengan Validasi Pengukuran Menggunakan Ray Tracing.

| Identifikasi Lintasan | | DOA (°) | Eror (DOA) | DOD (°) | Eror (DOD) | Delay (ns) | Panjang Lintasan (m) |
|-----------------------|-------------|------------|---------------|------------|---------------|---------------|-------------------------|
| LOS (#1) | Ray Tracing | 90 | 35 | 90 | 9 | 0 | 10.5 |
| | SAGE | 125 | | 99 | | 0 | |
| #2 (C,A) | Ray Tracing | 90 | 29 | 90 | 19 | 92.66 | 38.3 |
| | SAGE | 119 | | 109 | | 90 | |
| #3 (tiang, B) | Ray Tracing | 173 | 46 | 7.1 | 36.9 | 249.166 | 85.25 |
| | SAGE | 127 | | 44 | | 240 | |

Tabel 4.2 menunjukkan bahwa lintasan #1 merupakan komponen LOS dengan daya impuls tertinggi dengan excess delay 0 ns. Hasil Ray tracing menunjukkan bahwa DOA dan DOD terdeteksi 90°, sedangkan hasil SAGE mendeteksi DOA dan DOD untuk lintasan LOS berturut turut 125° dan 99°. Lintasan #2 yang merupakan lintasan dari dinding C ke dinding A berdasarkan ray tracing terdeteksi DOA dan DOD berturut-turut adalah 90° dan 90°, sedangkan hasil SAGE berturut-turut adalah 119° dan 109°. Lintasan #3 yang merupakan lintasan dari tiang ke dinding B terdeteksi 173° dan 7.1° sedangkan hasil SAGE berturut-turut adalah 127° dan 44°.

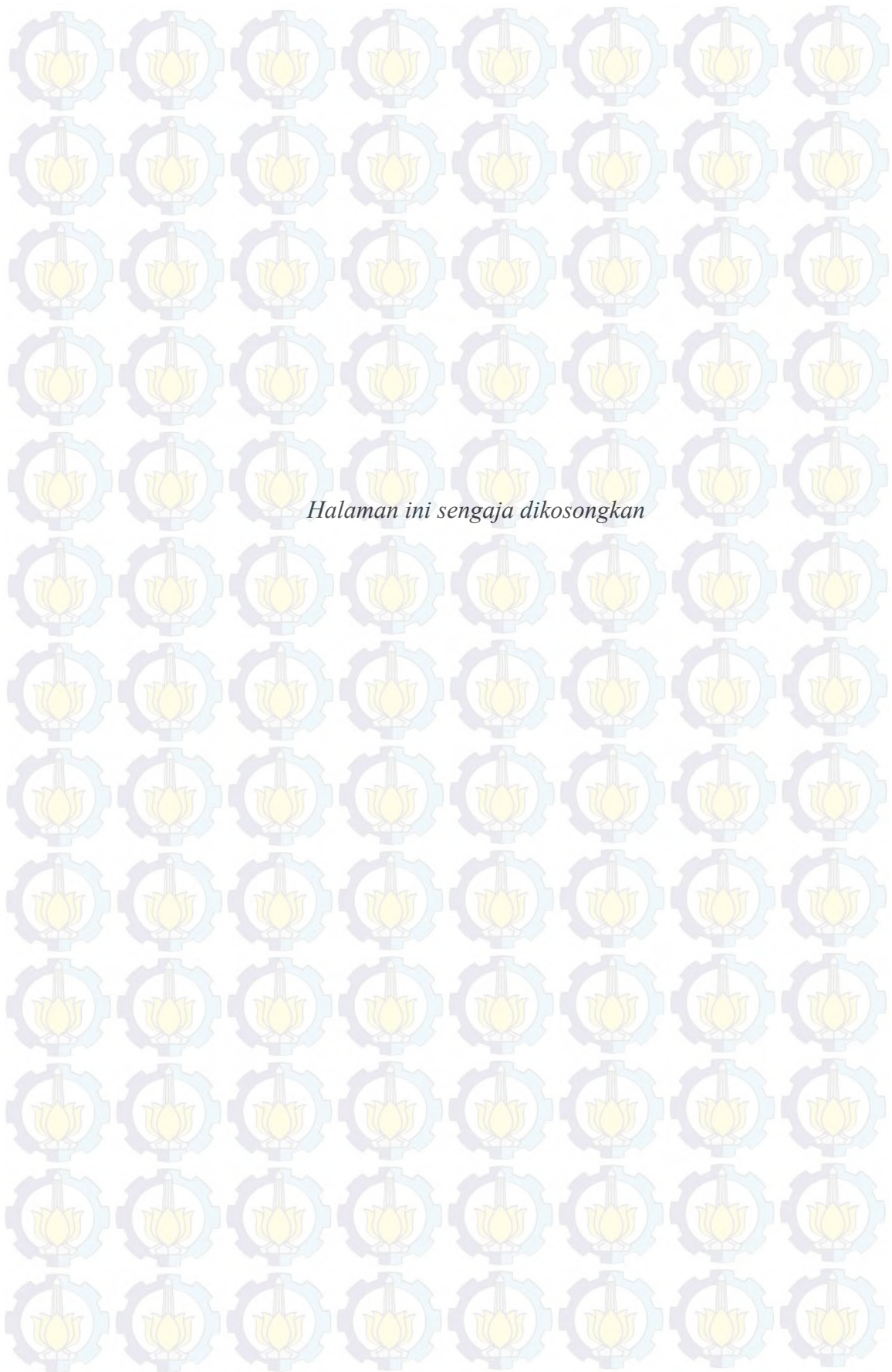
Penyimpangan hasil validasi pengukuran dengan hasil estimasi disebabkan oleh beberapa faktor yaitu sumber sinyal yang datang dari perangkat wifi yang lain, meskipun telah dilakukan pengecekan frekuensi wifi sebelum pengukuran dilakukan atau mencari channel yang kosong dengan wifi analyzer, sehingga memungkinkan frekuensi wifi lain atau hot spot yang mengganggu frekuensi pengukuran. Faktor yang kedua adalah disebabkan oleh human eror, dan faktor yang ketiga disebabkan oleh pola radiasi dari antena linier.

Berdasarkan tabel 4.2 terlihat bahwa sudut DOA mengalami penyimpangan (kesalahan antara validasi pengukuran dan hasil estimasi) yang lebih besar dibandingkan dengan penyimpangan yang disebabkan oleh DOA, hal inipun juga

telah terbukti secara simulasi bahwa estimasi DOD lebih baik dibandingkan dengan DOA.

Selain itu untuk lintasan #3 terjadi penyimpangan terbesar, hal ini disebabkan oleh pola radiasi dari antena array linier, dimana apabila terdapat sinyal yang datang mendekati 0° dan 180° mengalami penurunan gain sehingga hasil estimasi SAGE mengalami kesalahan atau penyimpangan yang besar.

Selain itu hasil dari validasi tersebut mengalami kesalahan karena keterbatasan alat dengan resolusi yang rendah sehingga mempengaruhi hasil estimasi SAGE, akibat resolusi tersebut membuat terjadi penjumlahan sinyal atau terdapat komponen sinyal yang numpuk sehingga SAGE tidak bisa mendeteksi penumpukan sinyal tersebut, karena SAGE mendeteksi per komponen dari sinyal yang diterima. Hasil identifikasi sinyal tersebut terdeteksi sebanyak 13 lintasan (dapat dilihat pada tabel lampiran), dari 13 lintasan tersebut berasal dari satu pantulan, dua pantulan. Dari sekian lintasan tersebut yang terdeteksi oleh SAGE adalah lintasan LOS lintasan akibat dua pantulan yang bisa dilihat pada tabel 4.2.



BAB 5

PENUTUP

Pada bab ini akan diuraikan beberapa kesimpulan yang dapat diambil dari pembahasan sebelumnya dan saran mengenai masalah yang bisa dibahas sebagai kelanjutan penelitian ini.

5.1 Kesimpulan

Berdasarkan hasil simulasi, Pengukuran, Analisa data dan Pembahasan yang dilakukan dalam penelitian ini, ada beberapa kesimpulan yang dapat diberikan sebagai berikut :

1. Secara simulasi algoritma SAGE dapat melakukan estimasi parameter kanal (delay, DOA, DOD, dan amplitudo kompleks), yang dibuktikan dengan validasi simulasi berdasarkan metode analisa simulasi yaitu perubahan SNR, perubahan jumlah path, dan perubahan beda sudut DOA dan DOD. Selain itu, simulasi yang diawali dengan deteksi delay (SAGE1) maupun simulasi yang diawali dengan deteksi sudut (SAGE2) mempunyai kinerja yang berbeda untuk setiap metode analisa simulasi yang digunakan pada antenna array linier dan antenna array planar.
 - a. Hasil estimasi parameter delay dan parameter sudut terhadap variasi SNR 0-30 dB menunjukkan semakin besar nilai SNR, maka nilai RMS eror mendekati 0 ns pada simulasi SAGE1 dan SAGE2 untuk antenna array planar maupun antenna array linier, sedangkan hasil estimasi parameter amplitudo kompleks menunjukkan bahwa semakin besar nilai SNR maka nilai Rasio (dB) antara hasil estimasi amplitudo kompleks dengan hasil input amplitudo kompleks, nilai mendekati 0 dB.
 - b. Hasil estimasi parameter delay dan sudut terhadap variasi jumlah multipath pada simulasi SAGE1 dan SAGE2 untuk antenna array planar maupun linier menunjukkan semakin besar jumlah multipath semakin besar nilai RMS eror yang didapatkan. Sedangkan hasil estimasi parameter amplitudo kompleks menunjukkan bahwa semakin besar jumlah path , Rasio amplitudo terestimasi terhadap amplitudo sesungguhnya (dB) yang terjadi akibat multipath tersebut semakin besar.

- c. Saat diterapkan untuk estimasi beda sudut DOA dan DOD untuk antenna array planar, membuktikan bahwa simulasi SAGE1 dapat membedakan sudut DOA yang sama atau sudut yang berjauhan dengan RMS error tertinggi pada 4.3° , sedangkan SAGE2 dapat membedakan sudut DOA dengan RMS error tertinggi 0.9° . sedangkan untuk estimasi beda sudut DOD pada algoritma SAGE1 nilai RMS error tertinggi pada 1.3° , sedangkan pada algoritma SAGE2 nilai RMS error tertinggi pada 0.8° .
- d. Saat diterapkan untuk estimasi beda sudut DOA dan DOD untuk antenna array linier, membuktikan bahwa simulasi SAGE1 dapat membedakan sudut DOA yang sama atau sudut yang berjauhan dengan RMS error tertinggi pada 1° , sedangkan SAGE2 dapat membedakan sudut DOA dengan RMS error tertinggi 1.8° . sedangkan untuk estimasi beda sudut DOD pada algoritma SAGE1 nilai RMS error tertinggi pada 2.8° , sedangkan pada algoritma SAGE2 nilai RMS error tertinggi pada 0.8° .
- e. Hasil estimasi perubahan sudut elevasi DOA dan DOD pada antenna array linier mempengaruhi kinerja SAGE1 maupun SAGE2, untuk sudut elevasi DOA menuju 0 dan menuju 180 nilai RMS error tertinggi 11° , sedangkan untuk sudut elevasi DOD nilai RMS error tertinggi 12.5° hal ini terjadi pelemahan gain dari pola radiasi antenna array linier.
- f. Hasil estimasi sudut DOA dan DOD untuk komponen yang mempunyai delay yang sama nilai RMS error tertinggi 0.7° untuk antenna array planar maupun antenna array linier.

Dari kesimpulan setiap pengamatan di atas, dapat disimpulkan bahwa simulasi SAGE2 lebih direkomendasikan dibandingkan dengan simulasi SAGE1 pada antenna array planar maupun antenna array linier.

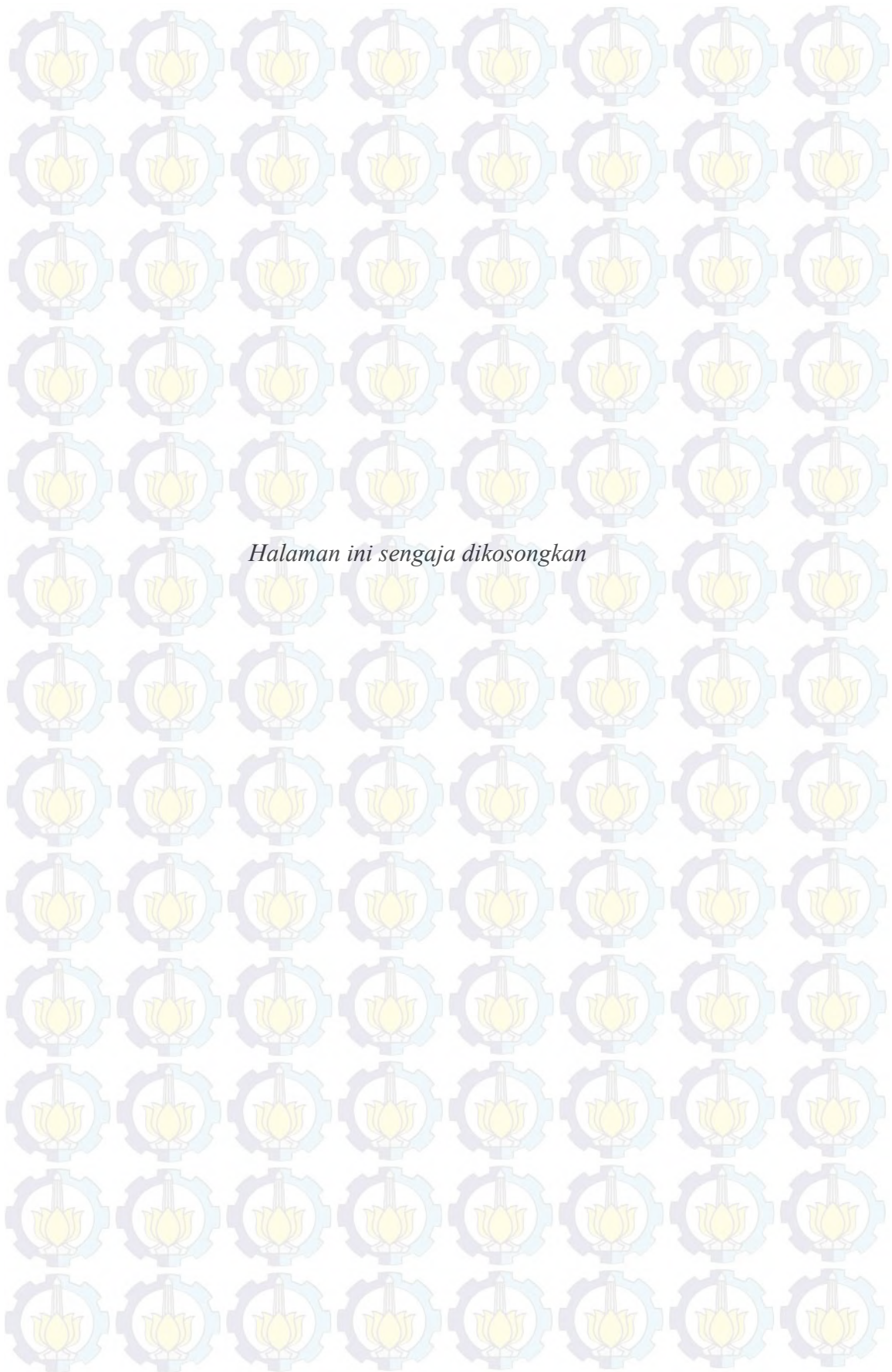
2. Validasi lintasan sinyal dilakukan dengan menggunakan Ray Tracing 2D, metode tersebut dilakukan dengan cara melakukan pencerminan posisi array TX dan RX 3 bidang pencerminan di Gedung Hall D4 PENS, sehingga didapatkan sudut DOA, DOD dan panjang lintasan/time delay sinyal lintasan jamak pada lingkungan pengukuran. Algoritma SAGE hanya bisa mendeteksi tiga komponen sinyal, yaitu satu komponen dari

lintasan LOS dan 2 komponen dari dua pantulan, sedangkan lintasan satu pantulan tidak terdeteksi oleh SAGE dengan baik, Hal ini dikarenakan resolusi perangkat yang rendah sehingga mengakibatkan komponen yang tertumpuk yang mengakibatkan SAGE tidak bisa mendeteksi komponen sinyal yang tertumpuk tersebut.

5.2 SARAN

Berdasarkan hasil analisa simulasi dan validasi pengukuran yang telah dipaparkan pada Bab IV, dapat diberikan beberapa saran yang berguna untuk pengembangan penelitian ini, Berikut adalah saran yang dapat diberikan :

1. Kinerja SAGE mendeteksi parameter sudut dulu harus dikembangkan lagi, apalagi diterapkan pada sistem pengukuran dengan resolusi kecil (≥ 60 ns). Sehingga dibutuhkan teknik preprocessing data pengukuran sebelum diproses menggunakan SAGE.
2. Sistem pengukuran harus dirancang dengan baik untuk mengatasi keterbatasan resolusi alat USRP, misalnya menggunakan tambahan tx *reference* untuk mengatasi ketidaksinkronan antara USRP sebagai TX dan RX.



Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Bernard, Fleury. "Channel Parameter Estimation in Mobile Radio Environments Using the SAGE Algorithm," IEEE Journal On Selected Areas On Communication , vol.17, pp.434-450, March 1999.
- [2] Syahroni, M., (2010), "Estimasi DOA, DOD, dan Delay Menggunakan Algoritma MUSIC pada Antena Array Volume", Thesis Magister , Program Studi Teknik Elektro Pasca Sarjana ITS.
- [3] Handayani, P., (2013) "Model Spasial-Temporal Kanal Radio Dua Arah untuk Sistem Komunikasi Multiantena Kooperatif dalam Gedung pada pita 2.5GHz", Disertasi Doktor Program Studi Teknik Elektro Pasca Sarjana ITS.
- [4] Yasdinul, H., (2010), "Estimasi Parameter Kanal : Delay, DoA, DoD, dengan Algoritma SAGE Pada Antena Array Volume", Thesis Magister , Program Studi Teknik Elektro Pasca Sarjana ITS.
- [5] Chong C.C, Laurenson D.I., Tan C.M., Mc Laughlin S., Beach M.A., Nix A.R. "Joint Detection-Estimation of Directional Channel Parameter Using the 2-D Frekuensi Domain SAGE Algorithm with Serial Interference Cancellation", IEEE-0-7803-7400-2/2002
- [6] Wyne S., Molisch A.F., Almers P., Eriksson G. (2005), "Statistical Evaluation of Outdoor to Indoor Office MIMO Measurements at 5.2 GHz" IEEE-0-7803-8887-9/2005.
- [7] Nayerlaan, J. D, (1999) , Spread Spectrum (SS) Introduction, Sirius Communications
- [8] Stutzman,L.W, dan Tiele G.A, (2013) , Antenna Theory and Design, 3rd edition, Wiley, United States.
- [9] Fleury, H.B, "High Resolution Channel Parameter Estimation for MIMO Applications Using the SAGE Algorithm", IEEE International Zurich Seminar on Broadband Communications, 2002.

[10] Yang C.S , (1998), *CDMA RF System Engineering*, Artech House, Boston, London

[11] Molisch, A, (2011), *Wireless Communication Second Edition*, Wiley IEEE, California, USA

[12] Huda Y, Syahroni, Hadayani, “*Evaluasi Sistem Pengukuran Kanal Radio Dua Arah Dalam gedung Pita 2.5 GHz Menggunakan Antena Larik 3 Dimensi*”, ITS, 2010

[13] Yang C.S , (1998), *CDMA RF System Engineering*, Artech House, Boston, London

[14] Sousa, E. S, Javanovic, V. M., Daigneault, C., (1994), “Delay Spread Measurement for Digital Cellular Channel in Toronto”, IEEE Transaction on Vehicular Technology, Vol.43, No.4, hal. 837-847

LAMPIRAN

Lampiran terdiri dari program simulasi dan data simulasi serta data pendukung yang digunakan pada pengukuran.

1. Simulasi SAGE1

```
clc; clear all; close all;
disp('-----');
disp('    Silahkan memasukkan parameter kanal');
disp('-----');
M = 2; % rx antennas
N = 4; % Nr x Nt
Nt = 2; % TX antenn
Nr = 2; % RX antenna
c = 3*(10^8); % velocity of light
f = 2.425e9; %Frequency MHZ
lamda = c/(f);
a = 1/2;
r = a*lamda; % space between antennas
P = [0 0 0; 0 0 r]; % potition of Antenna with array linier
P = [0 0 0; r 0 0; r r 0; 0 r 0]; % potition of Antenna array
planar
beta = 2*pi/lamda; % beta
load str_linier;
Str_vctr = str_linier;
e11 = 0; e12 = 180;

L = input('Silakan inputkan jumlah sinyal yang akan diestimasi = ');
for jml = 1:L
    disp(['PARAMETER SINYAL KE ' num2str(jml)]);
    gdelay(jml) = input ('input delay = ');
    gelev2 (jml) = input ('input elevasi doa = ');
    gelev1(jml) = input ('input elevasidod = ');
end
%-----PN Code generation-----
m = 8;
p_code = 2^(m)-1;
code_awal = [1 0 0 0 1 1 1 0]; % lihat tabel [7,1]
for k = 1: p_code
    temp(k) =
    mod(code_awal(8)+code_awal(4)+code_awal(3)+code_awal(3),2);
    code_awal = [temp(k) code_awal(1:m-1)];
end
temp2 = [temp; temp];
temp2_reshape = reshape(temp2,1,[]); % 1 x 510
sinyal_info = ones (L,length(temp2_reshape));
sudut = randn(2,length(sinyal_info)).*(2*pi);
SS = [];
for i = 1:L
    sinyal_tx = mod(temp2_reshape + sinyal_info(i,:),2);
    sinyal_mod = (sinyal_tx.*2) -1;
    sinyal_terphase = sinyal_mod.*exp(j*sudut(i,:));
    sinyal_kirim = [SS ; sinyal_terphase];
```



```

SS = sinyal_kirim;
end
bpsk = sinyal_kirim; % L x K
K = length(bpsk);
% -----
--
%amplitudo yang diatur = 1
a = randn(L,1) + j*randn(L,1);
a = a./abs(a); alfa = a; % L x 1
% get steering vector for DOA and DOD
steering_vector_doa = sv_linier (gelev2(1:L)); % Nr x L
steering_vector_dod = sv_linier (gelev1(1:L)); % Nr x L
%-----delayed signal-----%
for I = 1: L
    h = dfilt.delay (gdelay(I));
    delay_awal = filter (h,bpsk(I,:));
    ut(I,:) = delay_awal; % L x K
end

%-----Noise (AWGN)-----%
snr = 30; % dB
noise = randn(N,K) + j*randn (N,K);
noisel1 = noise./(abs (noise));
noisel2 = noisel1 *(1/(10^(snr/10))); % N x K
%-----%

%----- Output signal Array -----
Nn = 1;
for kk = 1: Nt
    for l = 1:Nr
        for Ln = 1:L
            Array(Ln,:) = alfa(Ln)* steering_vector_doa(l,Ln)*
steering_vector_dod(kk,Ln)*ut(Ln,:);
        end
        if L == 1
            Array2(Nn,1:K) = (Array); % untuk jumlah sinyal L = 1
        else L ~= 1
            Array2(Nn,1:K) = sum (Array); % N x K, untuk jumlah L
> 1
        end
        Nn = Nn+1;
    end
end
end
% -----
% ----- Noise + transmitted signal -----
Yt = Array2 + noisel2; % N x K
% -----
%=====

```

```

% =====*** 1. Pre-Initialitation Step ***=====
% =====
for n = 1:L
    sudut_estdoal(n) = 0 ;
    sudut_estdodl(n) = 0 ;
    amp_complex1(n) = 0 ;
    time_delay1(n) = 0;
    fprintf ('%d : %d : %d : %d \t', time_delay1(n),...
        sudut_estdoal(n), sudut_estdodl(n), amp_complex1(n));
end
fprintf ('\n');

% ----- TAHAP INISIALISASI -----
for miu = -(L-1):1:0
    el = miu + L;
    % ----- Proses E-STEP -----
    n = [];
    for Ndep = 1:Nt
        for Narr = 1:Nr
            d = zeros (1,K);
            for q = 1:L
                if q ~= el
                    str1(:,q) = sv_linier (sudut_estdoal(q));
                    str2(:,q) = sv_linier (sudut_estdodl(q));
                    h = dfilt.delay(time_delay1(q));
                    delay = filter (h,bpsk(q,:));
                    alfa1 (q,:) = delay;
                else
                    str1(:,el) = zeros (M,1);
                    str2(:,el) = zeros (M,1);
                    alfa1 (q,:) = zeros(1,K);
                end
                amp_ins = amp_complex1;
                sinyal =
                amp_ins(q)*str2(Narr,q)*str1(Ndep,q)*alfa1(q,:);
                s_n = d + sinyal;
                d = s_n;
            end %q
            sn_tot = [n; s_n];
            n = sn_tot;
        end % Narr
    end % Ndep
    % Proses dekomposisi
    X_ekspektasi = Yt - sn_tot; % N x K
    % ----- FINISH E-STEP -----

    % ----- Proses M-STEP -----
    % -----Estimasi Delay -----
    del_ins = K;
    haha = [];
    for jj = 0:1:(del_ins-1)
        h = dfilt.delay (jj);
        delay = filter (h,bpsk(el,:));
        z_del_1 = delay;
        z_dell = [haha ; z_del_1];
        haha = z_dell;
    end

```



```

[ab,mn] = size( z_dell);
cc = [] ; dd = [];
for k = 1:ab
    for p = 1:N
        X_Delay = X_ekspektasi(p,:).*conj(z_dell(k,:));
        X_Delay1 = [cc; X_Delay];
        cc = X_Delay1; % N x K
    end
    cc = [];
    abs_delay = abs(sum(X_Delay1,2).^2);
    Z_tot_delay = sum (abs_delay,1);
    z_te_delay1 = [dd ; Z_tot_delay];
    dd = z_te_delay1; % 124x1 --> dimana setiap sampel 124
    mewakili 20ns
end

% get max value from z_del_max
for del = 1 : ab
    if z_te_delay1(del,:) - max(z_te_delay1) == 0
        DELAY = (del-1);
    end
end
time_delay1(el)= DELAY; % delay estimation

% --- Estimasi DOA ---
h = dfilt.delay(time_delay1(el));
states = h.states;
delay = filter (h, bpsk(el,:));
Xn_tau_new = delay; % 1 x K

pp = [];
for m = 1:N
    X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
    X_ntau = [pp ; X_tau];
    pp = X_ntau;
end
Z_doa_del3 = reshape (X_ntau,Nr,Nt,[]); %Nr x Nt x K
X_ekspektasi2 = sum (Z_doa_del3,2);
X_ekspektasi3 = reshape (X_ekspektasi2,Nr,[]); % Nr x K
X_ekspektasi4 = sum (X_ekspektasi3,2); % Nr x 1

mm = [];
for ib = 1: length(Str_vctr)
    z_doa = Str_vctr(:,ib) '* X_ekspektasi4;
    Z_Doal = [mm ; z_doa];
    mm = Z_Doal; % p_str_vctr x Nr
end

oa = [];
for matx = 1:length (Str_vctr)
    norm_DOA = (norm(Z_Doal(matx)))^2;
    norm_zDOA = [oa; norm_DOA];
    oa = norm_zDOA;
end
max_Z_DOA = max (norm_zDOA);
for m_doa = e11:e12
    if norm_zDOA(m_doa+1,:) - max (norm_zDOA) == 0

```

```

        elev_doa = m_doa;
    end
end
sudut_estdoa1(e1) = elev_doa ;

% ----- Estimasi DOD -----
strvctr_doa = sv_linier(sudut_estdoa1(e1));
do = []; od = [];

for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doa_del3(:, :, ss)*conj (Str_vctr(:, s));
        z_tot1 = [do z_tot_dod];
        do = z_tot1;
    end
    sum_ztot1 = sum (z_tot1, 2);
    do = [];
    z_tot_1 = strvctr_doa' * sum_ztot1;
    z_tot_dod1 = [od z_tot_1];
    od = z_tot_dod1;
end

% normaliasasi ARg_max_dod dari fungsi korelasi
for matx = 1:length(Str_vctr)
    norm_ZDOD(matx, :) = (norm( z_tot_dod1(matx)).^2);
end

for m_dod = e11:e12
    if norm_ZDOD(m_dod+1, :) - max(norm_ZDOD) == 0
        elev_dod = m_dod;
    end
end
sudut_estdod1(e1) = elev_dod ; % get dod estimated

% Estimasi Amplitudo Kompleks
strvctr_dod = sv_linier (sudut_estdod1(e1));
za = [];
for t = 1:K
    X_amp = strvctr_doa'* Z_doa_del3(:, :, t)*conj(strvctr_dod);
    z_amp = [za X_amp];
    za = z_amp;
end
amp_complex1(e1) = (1/(Nr*Nt*K))*sum (z_amp); % get
amp_complex estimated
end
for n = 1:L
    fprintf ('%d : %d : %d : %d \t', time_delay1(n), ...
        sudut_estdoa1(n), sudut_estdod1(n), amp_complex1(n));
end
fprintf ('\n')

for V = 1:L-1
    konvergen(V) = 0;
end

```



```

% ===== Proses Update =====
for v = 1:L
    el = mod (miu,L) + 1;
    % cek konvergensi
    for V = 1:L-1
        while (konvergen(V) == el)
            time_delay(el) = time_delay1(konvergen(V));
            sudut_estdoa(el) = sudut_estdoa1(konvergen(V));
            sudut_estdod(el) = sudut_estdod1(konvergen(V));
            amp_complex(el) = amp_complex1(konvergen(V));
            if el == L
                for n = 1:L
                    fprintf ('%d : %d : %d : %d \t',
                        time_delay1(n), ...
                        sudut_estdoa1(n), sudut_estdod1(n), amp_complex1(n));
                end
                fprintf ('\n')
            end % el == L
            miu = miu + 1;
            el = mod (miu,L) + 1;
        end % while
    end % V

% melakukan E- STEP
% ----- Proses E-STEP -----
n = [];
for Ndep = 1:Nt
    for Narr = 1:Nr
        d = zeros (1,K);
        for q = 1:L
            if q ~= el
                str1(:,q) = sv_linier (sudut_estdoa1(q));
                str2(:,q) = sv_linier (sudut_estdod1(q));
                h = dfilt.delay(time_delay1(q));
                delay = filter (h, bpsk(q,:));
                alfa1 (q,:) = delay;
            else
                str1(:,el) = zeros (M,1);
                str2(:,el) = zeros (M,1);
                alfa1 (q,:) = zeros(1,K);
            end
            amp_ins = amp_complex1;
            sinyal =
                amp_ins(q)*str2(Narr,q)*str1(Ndep,q)*alfa1(q,:);
            s_n = d + sinyal;
            d = s_n;
        end %q
        sn_tot = [n; s_n];
        n = sn_tot;
    end % Narr
end % Ndep

% Proses dekomposisi
X_ekspektasi = Yt - sn_tot; % N x K
% ----- FINISH E-STEP -----

```

```

% ----- Proses M-STEP -----
% -----Estimasi delay -----
del_ins = K;
haha = [];
for jj = 0:1:(del_ins-1)
    h = dfilt.delay (jj);
    delay = filter (h,bpsk(el,:));
    z_del_1 = delay;
    z_dell = [haha; z_del_1];
    haha = z_dell;
end
[ab,mn] = size(z_dell);
cc = [] ; dd = [];
for k = 1:ab
    for p = 1:N
        X_Delay = X_ekspektasi(p,:).*conj(z_dell(k,:));
        X_Delay1 = [cc; X_Delay];
        cc = X_Delay1; % N x K
    end
    cc = [];
    abs_delay = abs(sum(X_Delay1,2).^2);
    Z_tot_delay = sum (abs_delay,1);
    z_te_delay1 = [dd ; Z_tot_delay];
    dd = z_te_delay1; % 1xK
end

% get max value from z_del_max
for del = 1 : ab
    if z_te_delay1(del,:) - max(z_te_delay1) == 0
        DELAY = (del-1);
    end
end
time_delay(el)= DELAY; % delay estimation

% untuk sinyal yang terdelay hasil update(delay")
h = dfilt.delay(time_delay(el));
states = h.states;
delay = filter (h,bpsk(el,:));
Xn_tau_new = delay; % 1 x K
pp=[];
for m = 1:N
    X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
    X_ntau = [pp ; X_tau];
    pp = X_ntau;
end
Z_doa_del3 = reshape (X_ntau,Nr,Nt,[]); %Nr x Nt x K

% --- Estimasi DOA ---
X_ekspektasi2 = sum (Z_doa_del3,2);
X_ekspektasi3 = reshape (X_ekspektasi2,Nr,[]); % Nr x K
X_ekspektasi4 = sum (X_ekspektasi3,2); % Nr x 4
mm = [];
for ib = 1: length(Str_vctr)
    z_doa = Str_vctr(:,ib) '* X_ekspektasi4;
    Z_Doal = [mm ; z_doa];
    mm = Z_Doal; % p_str_vctr x Nr
end

```



```

end
oa = [];
for matx = 1:length (Str_vctr)
    norm_DOA = (norm(Z_Doal(matx))).^2;
    norm_zDOA = [oa; norm_DOA];
    oa = norm_zDOA;
end
for m_doa = e11:e12
    if norm_zDOA (m_doa+1,:) - max(norm_zDOA) == 0
        elev_doa = m_doa;
    end
end
sudut_estdoa(e1) = elev_doa; % get doa estimated

% ----- Estimasi DOD -----
strvctr_doa = sv_linier (sudut_estdoa(e1));
do = []; od = [];
for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doa_del3(:, :, ss)*conj (Str_vctr(:,s));
        z_tot1 = [do z_tot_dod];
        do = z_tot1;
    end
    sum_ztot1 = sum (z_tot1,2);
    do = [];
    z_tot_1 = strvctr_doa' * sum_ztot1;
    z_tot_dod1 = [od z_tot_1];
    od = z_tot_dod1;
end
for matx = 1:length(Str_vctr)
    norm_ZDOD(matx,:) = (norm( z_tot_dod1(matx))).^2;
end
Arg_max_dod = max (norm_ZDOD);
for m_dod = e11:e12
    if norm_ZDOD(m_dod+1,:) - max(norm_ZDOD) == 0
        ele_dod = m_dod;
    end
end
sudut_estdod(e1) = elev_dod;
strvctr_dod = sv_linier (sudut_estdod(e1));

% -- estimasi amplitudo kompleks --
za = [];
for t = 1:K
    X_amp = strvctr_doa'* Z_doa_del3(:, :,t)*conj(strvctr_dod);
    z_amp = [za X_amp];
    za = z_amp;
end
amp_complex(e1) = (1/(Nr*Nt*K))*sum (z_amp);

% Menghitung error
sudut2 = sudut_estdoa(e1) - sudut_estdoa1(e1);
sudut1 = sudut_estdod(e1) - sudut_estdod1(e1);
amp = amp_complex(e1) - amp_complex1(e1);
time = time_delay(e1) - time_delay1(e1);

```

```

% Masuk ke Looping Besar (Pencapaian Konvergen)
while ((abs(sudut2)>0) || (abs(sudut1)>0) || (abs(amp)>10e-5)
|| (abs(time)>0))
    sudut_estdoa1(el) = sudut_estdoa(el);
    sudut_estdod1(el) = sudut_estdod(el);
    amp_complex1(el) = amp_complex(el);
    time_delay1(el) = time_delay(el);

    % cetak apabila el == L
    if el == L
        for n = 1:L
            fprintf ('%d : %d : %d : %d \t',
time_delay1(n),...
sudut_estdoa1(n),sudut_estdod1(n),amp_complex1(n));
        end
        fprintf ('\n')
    end
    miu = miu + 1;
    el = mod (miu,L) + 1;

    % Periksa Konvergensi lagi,
    for V = 1: L-1
        while (konvergen(V) == el)
            time_delay(el) = time_delay1(konvergen(V));
            sudut_estdoa(el) = sudut_estdoa1(konvergen(V));
            sudut_estdod(el) = sudut_estdod1(konvergen(V));
            amp_complex(el) = amp_complex1(konvergen(V));
            if el == L
                for n = 1:L
                    fprintf ('%d : %d : %d : %d \t',
time_delay1(n),...
sudut_estdoa1(n),sudut_estdod1(n),amp_complex1(n));
                end
                fprintf ('\n')
            end % el == L
            miu = miu + 1;
            el = mod (miu,L) + 1;
        end % while
    end % V
    % Selesai pemeriksaan konvergensi

    % Kembali ke E - STEP lagi
    n = [];
    for Ndep = 1:Nt
        for Narr = 1:Nr
            d = zeros (1,K);
            for q = 1:L
                if q ~= el
                    str1(:,q) = sv_linier (sudut_estdoa1(q));
                    str2(:,q) = sv_linier (sudut_estdod1(q));
                    h = dfilt.delay(time_delay1(q));
                    delay = filter (h,bpsk(q,:));
                    alfa1 (q,:) = delay;

```



```

else
    str1(:,el) = zeros (M,1);
    str2(:,el) = zeros (M,1);
    alfa1 (q,:) = zeros(1,K);
end
amp_ins = amp_complex1;
sinyal =
amp_ins(q)*str2(Narr,q)*str1(Ndep,q)*alfa1(q,:);
s_n = d + sinyal;
d = s_n;
end %q
sn_tot = [n; s_n];
n = sn_tot;
end % Narr
end % Ndep
% Proses dekomposisi
X_ekspektasi = Yt - sn_tot; % N x K
% --- Finish E- STEP ---

% Kembali ke Proses M-STEP lagi
% ----estimasi delay -----
del_ins = K; haha = [];
for jj = 0:1:(del_ins-1)
    h = dfilt.delay (jj);
    delay = filter (h,bpsk(el,:));
    z_del_1 = delay;
    z_dell = [haha; z_del_1]; % del_ins x K
    haha = z_dell;
end
[ab,mn] = size(z_dell);
cc = [] ; dd = [];
for k = 1:ab
    for p = 1:N
        X_Delay = X_ekspektasi(p,:).*conj(z_dell(k,:));
        X_Delay1 = [cc; X_Delay];
        cc = X_Delay1; % N x K
    end
    cc = [];
    abs_delay = abs(sum(X_Delay1,2).^2);
    Z_tot_delay = sum (abs_delay,1);
    z_te_delay1 = [dd ; Z_tot_delay];
    dd = z_te_delay1; % 1xK
end

% get max value from z_del_max
for del = 1 : ab
    if z_te_delay1(del,:) - max(z_te_delay1) == 0
        DELAY = (del-1);
    end
end
time_delay(el)= DELAY; % delay estimation
% terdelay dengan delay'
h = dfilt.delay(time_delay(el));
states = h.states;
delay = filter (h,bpsk(el,:));
Xn_tau_new = delay; % 1 x K
pp=[];

```

```

for m = 1:N
    X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
    X_ntau = [pp ; X_tau];
    pp = X_ntau;
end
Z_doa_del3 = reshape (X_ntau,Nr,Nt,[]); %Nr x Nt x K
% ---- Estimasi DOA ----
% menjumlah sebanyak antenna pengirim
X_ekspektasi2 = sum (Z_doa_del3,2);
X_ekspektasi3 = reshape (X_ekspektasi2,Nr,[]); % Nr x K
X_ekspektasi4 = sum (X_ekspektasi3,2); % Nr x 4
mm = [];
for ib = 1: length(Str_vctr)
    z_doa = Str_vctr(:,ib) '* X_ekspektasi4;
    Z_Doal = [mm ; z_doa];
    mm = Z_Doal; % p_str_vctr x Nr
end
oa = [];
for matx = 1:length (Str_vctr)
    norm_DOA = (norm(Z_Doal(matx))).^2;
    norm_zDOA = [oa; norm_DOA];
    oa = norm_zDOA;
end
for m_doa = e11:e12
    if norm_zDOA(m_doa+1,:) - max( norm_zDOA) == 0;
        elev_doa = m_doa;
    end
end
sudut_estdoa(e1) = elev_doa ; % get doa estimated

% ----- Estimasi DOD -----
strvctr_doa = sv_linier (sudut_estdoa(e1));
do = []; od =[];

for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doa_del3(:, :, ss)*conj
        (Str_vctr(:,s));
        z_tot1 = [do z_tot_dod];
        do = z_tot1;
    end
    sum_ztot1 = sum (z_tot1,2);
    do =[];
    z_tot_1 = strvctr_doa' * sum_ztot1;
    z_tot_dod1 = [od z_tot_1];
    od = z_tot_dod1;
end

for matx = 1:length(Str_vctr)
    norm_ZDOD(matx,:) = (norm( z_tot_dod1(matx))).^2;
end

for m_dod = e11:e12
    if norm_ZDOD(m_dod+1,:) - max (norm_ZDOD) == 0
        elev_dod = m_dod;
    end
end

```



```

end
sudut_estdod(e1) = elev_dod ; % get dod estimated
% ----- Estimasi Amplitudo Kompleks -----
strvctr_dod = sv_linier (sudut_estdod(e1));
za=[];
for t = 1:K
    X_amp = strvctr_doa'*
Z_doa_del3(:, :,t)*conj(strvctr_dod);
    z_amp = [za X_amp];
    za = z_amp;
end
amp_complex(e1) = (1/(Nr*Nt*K))*sum (z_amp);

% Menghitung Error
sudut2 = sudut_estdoa(e1) - sudut_estdoa1(e1);
sudut1 = sudut_estdod(e1) - sudut_estdod1(e1);
time = time_delay(e1) - time_delay1(e1);
amp = amp_complex(e1) - amp_complex1(e1);

end % while (...)
konvergen(v)= e1;
sudut_estdoa1(e1) = sudut_estdoa(e1);
sudut_estdod1(e1) = sudut_estdod(e1);
time_delay1(e1) = time_delay(e1);
amp_complex1(e1) = amp_complex(e1);
if e1 == L
    for n = 1:L
        fprintf ('%d : %d : %d : %d \t', time_delay1(n),...
            sudut_estdoa1(n), sudut_estdod1(n), amp_complex1(n));
    end
    fprintf ('\n')
end
miu = miu + 1;
end % v
% cetak hasil jika e1 ~= L
if e1 ~= L
    for n = 1:L
        fprintf ('%d : %d : %d : %d \t', time_delay1(n),...
            sudut_estdoa1(n), sudut_estdod1(n), amp_complex1(n));
    end
    fprintf ('\n')
end
% Mencetak hasil jika nilai e1 ~= L
for n = 1:L
    fprintf
    ('===== \n', n)
;
    fprintf ('Sinyal ke %d.\n', n );
    fprintf
    ('===== \n', n)
;
    fprintf ('Delay = %d \n', gdelay(n))
    fprintf ('Delay_estimasi = %d \n', time_delay1(n))
    fprintf ('DOA = %d \n', gelev2(n));
    fprintf ('DOA_estimasi = %d \n', sudut_estdoa1(n))

```

```

fprintf ('DOD = %d \n', gelev1(n))
fprintf ('DOD_estimasi = %d\n', sudut_estdod1(n))
fprintf ('Amp = %d \n', abs(alfa(n)));
fprintf ('Amp_estimasi = %d \n', abs(amp_complex1(n)));
end

```

2. Simulasi SAGE2

```

% --- BISMILLAH ---
% Program (DOA,DOD,Delay,Amplitudo Kompleks)
% SAGE2
clc; clear all; close all;
disp('-----');
disp('  Silahkan memasukkan parameter kanal');
disp('-----');
M = 2; % rx antennas
N = 4; % Nr x Nt
Nt = 2; % TX antenn
Nr = 2; % RX antenna
c = 3*(10^8); % velocity of light
f = 2.425e9; %Frequency MHZ
lamda = c/(f);
a = 1/2;
r = a*lamda; % space between antennas
P = [0 0 0; 0 0 r]; % potition of Antenna with array planar
snr = 30; % signal to noise r7atio
beta = 2*pi/lamda; % beta

load str_linier;
Str_vctr = str_linier;
el1 = 0; el2 = 180;

L = input('Silakan inputkan jumlah sinyal yang akan diestimasi = ');
for jml = 1:L
    disp(['PARAMETER SINYAL KE ' num2str(jml)]);
    gdelay(jml) = input ('input delay = ');
    gelev2 (jml) = input ('input elevasi doa = ');
    gelev1(jml) = input ('input elevasidod = ');
end

%-----PN Code generation-----
m = 8;
p_code = 2^(m)-1;
code_awal = [1 0 0 0 1 1 1 0]; % lihat tabel [7,1]
for k = 1: p_code
    temp(k) =
    mod(code_awal(8)+code_awal(4)+code_awal(3)+code_awal(3),2);
    code_awal = [temp(k) code_awal(1:m-1)];
end
temp2 = [temp; temp];
temp2_reshape = reshape(temp2,1,[]); % 1 x 510
sinyal_info = ones (L,length(temp2_reshape));
sudut = randn(2,length(sinyal_info)).*(2*pi);
SS = [];
for i = 1:L

```



```

sinyal_tx = mod(temp2_reshape + sinyal_info(i,:),2);
sinyal_mod = (sinyal_tx.*2) -1;
sinyal_terphase = sinyal_mod.*exp(j*sudut(i,:));
sinyal_kirim = [SS ; sinyal_terphase];
SS = sinyal_kirim;
end
bpsk = sinyal_kirim; % L x K
K = length(bpsk);
% -----
%amplitudo yang diatur = 1
a = randn(L,1) + j*randn(L,1);
a2 = a./abs(a); alfa = a2;
% get steering vector for DOA and DOD
steering_vector_doa = sv_linier (gelev2(1:L)); % Nr x L
steering_vector_dod = sv_linier (gelev1(1:L)); % Nr x L

%-----delayed signal-----%
for I = 1: L
    h = dfilt.delay (gdelay(I));
    delay_awal = filter (h,bpsk(I,:));
    ut(I,:) = delay_awal; % L x K
end
%-----Noise (AWGN)-----%
noise = randn(N,K) + j*randn (N,K);
noisel = abs (noise);
noise2 = (noise./noisel) *(1/(10^(snr/10))); % N x K
%-----%
%----- Output signal Array -----
Nn = 1;
for kk = 1: Nt
    for l = 1:Nr
        for Ln = 1:L
            Array(Ln,:) = alfa(Ln)* steering_vector_doa(l,Ln)*
steering_vector_dod(kk,Ln)*ut(Ln,:);
        end
        if L == 1
            Array2(Nn,1:K) = (Array);
        else
            Array2(Nn,1:K) = sum (Array); % N x K
        end
        Nn = Nn+1;
    end
end
end
% -----
% ----- Noise + transmitted signal -----
Yt = Array2 + noise2 ; % N x K
% -----
% =====*** 1. Pre-Initialitation Step **** =====
% =====
for n = 1:L
    sudut_estdoa1(n) = 0;
    sudut_estdod1(n) = 0;

```

```

amp_complex1(n) = 0 ;
time_delay1(n) = 0;
fprintf ('%d : %d : %d : %d\n', sudut_estdoal(n), sudut_estdod1(n),
time_delay1(n), amp_complex1(n));
end
fprintf ('\n');

% ==== Proses Inisialisasi =====
for miu = -(L-1):1:0
    el = miu + L;
    % ----- E-STEP -----
    n = [];
    for Ndep = 1:Nt
        for Narr = 1:Nr
            d = zeros (1,K);
            for q = 1:L
                if q ~= el
                    str1(:,q) = sv_linier (sudut_estdoal(q));
                    str2(:,q) = sv_linier (sudut_estdod1(q));
                    h = dfilt.delay(time_delay1(q));
                    delay = filter (h,bpsk(q,:));
                    alfa1 (q,:) = delay;
                else
                    str1(:,el) = zeros (M,1);
                    str2(:,el) = zeros (M,1);
                    alfa1 (q,:) = zeros(1,K);
                end
                amp_ins = amp_complex1;
                sinyal =
                    amp_ins(q)*str1(Narr,q)*str2(Ndep,q)*alfa1(q,:);
                s_n = d + sinyal;
                d = s_n;
            end %q
            sn_tot = [n; s_n];
            n = sn_tot;
        end % Narr
    end % Ndep
    % Proses dekomposisi
    X_ekspektasi = Yt - sn_tot; % N x K
    % ----- Finish E-STEP -----

    X_ekpekstasi2 = reshape (X_ekspektasi,Nr,Nt,K); % Nr x Nt x K
    X_ekpekstasi3 = sum (X_ekpekstasi2,2); % Nr x 1 x K
    X_ekpekstasi4 = reshape (X_ekpekstasi3,Nr,K); % Nr x K

    % ----- Proses M-STEP -----
    % 1.Estimasi DoA
    mm = [];
    for ib = 1: length(Str_vctr)
        z_doa = Str_vctr(:,ib)'* X_ekpekstasi4;
        norm_DOA = (norm(z_doa)).^2;
        Z_Doal = [mm ; norm_DOA];
        mm = Z_Doal; % p_str_vctr x 1
    end
    for m_doa = el1:el2
        if Z_Doal(m_doa+1,:) - max(Z_Doal) == 0

```



```

        elev_doa = m_doa;
    end
end
sudut_estdoal(el) = elev_doa; % get doa estimated

%2.Estimasi DoD
strvctr_doa = sv_linier (sudut_estdoal(el));
do = []; od = [];

gh = [];
for g = 1:K
    Z_DOA = strvctr_doa' * X_ekpekstasi2(:, :, g);
    Z_doA = [gh ; Z_DOA];
    gh = Z_doA; % K x Nt
end
for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doA(ss,:) * conj(Str_vctr(:,s)) ;
        z_tot1 = [do z_tot_dod];
        do = z_tot1; % 1 x K
    end
    sum_ztot1 = (norm (z_tot1)).^2;
    norm_Z_DOD = [od ; sum_ztot1];
    od = norm_Z_DOD;
    do = [];
end
for m_dod= el1:el2
    if norm_Z_DOD(m_dod+1,:) - max(norm_Z_DOD) == 0
        elev_dod = m_dod;
    end
end
sudut_estdod1(el) = elev_dod;
strvctr_dod = sv_linier (sudut_estdod1(el));

%3.estimasi delay
za = [];
for t = 1:K
    X_DEL = strvctr_doa' *
        X_ekpekstasi2(:, :, t) * conj(strvctr_dod);
    Z_DEL = [za X_DEL];
    za = Z_DEL;
end
del_ins = K;
for jj = 0:1:(del_ins-1)
    h = dfilt.delay (jj);
    delay = filter (h, bpsk(el, :));
    z_del_1 = delay;
    z_dell(jj+1, :) = z_del_1;
end
dd = [];
for k = 1:del_ins
    X_delay = Z_DEL .* conj (z_dell(k, :));
    abs_delay = abs(sum(X_delay, 2).^2);
    Z_t_delay = [dd; abs_delay];
    dd = Z_t_delay;
end
for del = 1 : del_ins

```

```

        if Z_t_delay(del,:) - max(Z_t_delay) == 0
            DELAY = del-1;
        end
    end
    time_delay1(el)= DELAY;
    h = dfilt.delay(time_delay1(el));
    states = h.states;
    delay = filter (h, bpsk(el,:));
    Xn_tau_new = delay; % 1 x K

    pp = [];
    for m = 1:N
        X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
        X_ntau = [pp ; X_tau];
        pp = X_ntau; % N x K
    end
    X_ntau2 = reshape (X_ntau,Nr,Nt,K);
    % 4. Estimasi amplitudo kompleks
    AMP = [];
    for t = 1:K
        X_amp = strvctr_doa'* X_ntau2(:, :,t)*conj(strvctr_dod);
        z_AMP = [AMP X_amp];
        AMP = z_AMP;
    end
    amp_complex1(el) = (1/(Nr*Nt*K))*sum (z_AMP);
end % miu
% print the output of initial step
for n = 1:L
    fprintf ('%d : %d : %d : %d\n',sudut_estdoa1(n),sudut_estdod1(n),
    time_delay1(n),amp_complex1(n));
end
fprintf ('\n')

for V = 1:L-1
    konvergen(V) = 0;
end

% ===== Proses Update =====
for v = 1:L
    el = mod(miu,L)+1;
    % cek konvergensi tahap 1
    for V = 1:L-1
        while (konvergen(V) == el)
            sudut_estdoa(el) = sudut_estdoa1(konvergen(V));
            sudut_estdod(el) = sudut_estdod1(konvergen(V));
            time_delay(el) = time_delay1(konvergen(V));
            amp_complex(el) = amp_complex1(konvergen(V));
            if el == L
                for n = 1:L
                    fprintf ('%d : %d : %d : %d\n',sudut_estdoa1(n),sudut_estdod1(n),
                    time_delay1(n),amp_complex1(n));
                end
                fprintf ('\n')
            end % el == L
            miu = miu + 1;
        end
    end
end

```



```

        e1 = mod (miu,L) + 1;
    end % while
end % V

% ----- E-STEP -----
n = [];
for Ndep = 1:Nt
    for Narr = 1:Nr
        d = zeros (1,K);
        for q = 1:L
            if q ~= e1
                str1(:,q) = sv_linier (sudut_estdoa1(q));
                str2(:,q) = sv_linier (sudut_estdod1(q));
                h = dfilt.delay(time_delay1(q));
                delay = filter (h,bpsk(q,:));
                alfa1 (q,:) = delay;
            else
                str1(:,e1) = zeros (M,1);
                str2(:,e1) = zeros (M,1);
                alfa1 (q,:) = zeros(1,K);
            end
            amp_ins = amp_complex1;
            sinyal =
            amp_ins(q)*str1(Narr,q)*str2(Ndep,q)*alfa1(q,:);
            s_n = d + sinyal;
            d = s_n;
        end %q
        sn_tot = [n; s_n];
        n = sn_tot;
    end % Narr
end % Ndep
% Proses dekomposisi
X_ekspektasi = Yt - sn_tot; % N x K
% --- Finish E-STEP---
X_ekpekstasi2 = reshape (X_ekspektasi,Nr,Nt,K); % Nr x Nt x K
X_ekpekstasi3 = sum (X_ekpekstasi2,2); % Nr x 1 x K
X_ekpekstasi4 = reshape (X_ekpekstasi3,Nr,K); % Nr x K

% ----- Proses M-STEP -----
% 1. Estimasi DoA
mm = [];
for ib = 1: length(Str_vctr)
    z_doa = Str_vctr(:,ib)'* X_ekpekstasi4;
    norm_DOA = (norm(z_doa)).^2;
    Z_Doal = [mm ; norm_DOA];
    mm = Z_Doal; % p_str_vctr x 1
end
for m_doa = e11:e12
    if Z_Doal(m_doa+1,:) - max(Z_Doal) == 0
        elev_doa = m_doa;
    end
end
sudut_estdoa(e1) = elev_doa;
% 2. Estimasi DoD
strvctr_doa = sv_linier (sudut_estdoa(e1));
do = []; od = [];
gh = [];

```

```

for g = 1:K
    Z_DOA = strvctr_doa' * X_ekpekstasi2(:, :, g);
    Z_doA = [gh ; Z_DOA];
    gh = Z_doA; % K x Nt
end
for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doA(ss,:) * conj(Str_vctr(:,s)) ;
        z_tot1 = [do z_tot_dod];
        do = z_tot1; % 1 x K
    end
    sum_ztot1 = (norm (z_tot1)).^2;
    norm_Z_DOD = [od ; sum_ztot1];
    od = norm_Z_DOD;
    do = [];
end
for m_dod= el1:el2
    if norm_Z_DOD(m_dod+1,:) - max(norm_Z_DOD) == 0
        elev_dod = m_dod;
    end
end
sudut_estdod(el) = elev_dod;
strvctr_dod = sv_linier (sudut_estdod(el));
% 3. Estimasi Delay
za = [];
for t = 1:K
    X_DEL = strvctr_doa' *
    X_ekpekstasi2(:, :, t) * conj(strvctr_dod);
    Z_DEL = [za X_DEL];
    za = Z_DEL; % 1 x K
end
% 4. Estimasi Amplitudo Kompleks
del_ins = K;
for jj = 0:1:(del_ins-1)
    h = dfilt.delay (jj);
    delay = filter (h, bpsk(el, :));
    z_del_1 = delay;
    z_dell(jj+1, :) = z_del_1;
end
dd = [];
for k = 1:del_ins
    X_delay = Z_DEL .* conj (z_dell(k, :));
    abs_delay = abs(sum(X_delay, 2).^2);
    Z_t_delay = [dd; abs_delay];
    dd = Z_t_delay;
end
for del = 1 : del_ins
    if Z_t_delay(del, :) - max(Z_t_delay) == 0
        DELAY = del-1;
    end
end
time_delay(el) = DELAY;
h = dfilt.delay(time_delay(el));
states = h.states;
delay = filter (h, bpsk(el, :));
Xn_tau_new = delay; % 1 x K
pp = [];
for m = 1:N

```



```

        X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
        X_ntau = [pp ; X_tau];
        pp = X_ntau; % N x K
    end
    X_ntau2 = reshape (X_ntau,Nr,Nt,K);
    AMP = [];
    for t = 1:K
        X_amp = strvctr_doa'* X_ntau2(:,:,t)*conj(strvctr_dod);
        z_AMP = [AMP X_amp];
        AMP = z_AMP;
    end
    amp_complex(el) = (1/(Nr*Nt*K))*sum (z_AMP); % get amp_complex
    estimated

% Menghitung Error
sudut2 = sudut_estdoa(el) - sudut_estdoa1(el);
sudut1 = sudut_estdod(el) - sudut_estdod1(el);
time = time_delay(el) - time_delay1(el);
amp = amp_complex(el) - amp_complex1(el);

%"""" Masuk Ke LOOP BESAR""""
while ((abs(sudut2)>0) || (abs(sudut1)>0) || (abs(amp)>10e-5)
|| (abs(time)>0))
    sudut_estdoa1(el) = sudut_estdoa(el);
    sudut_estdod1(el) = sudut_estdod(el);
    amp_complex1(el) = amp_complex(el);
    time_delay1(el) = time_delay(el);

    % cetak apabila el == L
    if el == L
        for n = 1:L
            fprintf ('%d : %d : %d : %d\n',
\t',sudut_estdoa1(n),sudut_estdod1(n),
time_delay1(n),amp_complex1(n));
        end
        fprintf ('\n')
    end
    miu = miu + 1;
    el = mod (miu,L) + 1;

% Periksa Konvergensi lagi,
for V = 1: L-1
    while (konvergen(V) == el)
        time_delay(el) = time_delay1(konvergen(V));
        sudut_estdoa(el) = sudut_estdoa1(konvergen(V));
        sudut_estdod(el) = sudut_estdod1(konvergen(V));
        amp_complex(el) = amp_complex1(konvergen(V));
        if el == L
            for n = 1:L
                fprintf ('%d : %d : %d : %d\n',
\t',sudut_estdoa1(n),sudut_estdod1(n),
time_delay1(n),amp_complex1(n));
            end
            fprintf ('\n')
        end % el == L
        miu = miu + 1;
    end
end

```

```

        el = mod (miu,L) + 1;
    end % while
end % V
% Selesai pemeriksaan konvergensi
% ---- E-STEP ----
n = [];
for Ndep = 1:Nt
    for Narr = 1:Nr
        d = zeros (1,K);
        for q = 1:L
            if q ~= el
                str1(:,q) = sv_linier (sudut_estdoal(q));
                str2(:,q) = sv_linier (sudut_estdodl(q));
                h = dfilt.delay(time_delay1(q));
                delay = filter (h,bpsk(q,:));
                alfa1 (q,:) = delay;
            else
                str1(:,el) = zeros (M,1);
                str2(:,el) = zeros (M,1);
                alfa1 (q,:) = zeros(1,K);
            end
            amp_ins = amp_complex1;
            sinyal =
            amp_ins(q)*str1(Narr,q)*str2(Ndep,q)*alfa1(q,:);
            s_n = d + sinyal;
            d = s_n;
        end %q
        sn_tot = [n; s_n];
        n = sn_tot;
    end % Narr
end % Ndep
% Proses dekomposisi
X_ekspektasi = Yt - sn_tot; % N x K
% ---- Finish E-STEP----
X_ekpekstasi2 = reshape (X_ekspektasi,Nr,Nt,K); % Nr x Nt x K
X_ekpekstasi3 = sum (X_ekpekstasi2,2); % Nr x 1 x K
X_ekpekstasi4 = reshape (X_ekpekstasi3,Nr,K); % Nr x K
% 1. Estimasi DoA
mm = [];
for ib = 1: length(Str_vctr)
    z_doa = Str_vctr(:,ib)' * X_ekpekstasi4;
    norm_DOA = (norm(z_doa)).^2;
    Z_Doal = [mm ; norm_DOA];
    mm = Z_Doal; % p_str_vctr x 1
end
for m_doa = el1:el2
    if Z_Doal(m_doa+1,:) - max(Z_Doal) == 0
        elev_doa = m_doa;
    end
end
sudut_estdoal(el) = elev_doa;
%2. Estimasi DoD
strvctr_doa = sv_linier (sudut_estdoal(el));
do = []; od = [];
gh = [];
for g = 1:K

```



```

Z_DOA = strvctr_doa' * X_ekpekstasi2(:, :, g);
Z_doA = [gh ; Z_DOA];
gh = Z_doA; % K x Nt
end
for s = 1: length (Str_vctr)
    for ss = 1:K
        z_tot_dod = Z_doA(ss, :) * conj(Str_vctr(:, s)) ;
        z_tot1 = [do z_tot_dod];
        do = z_tot1; % 1 x K
    end
    sum_ztot1 = (norm (z_tot1)).^2; % 4x1
    norm_Z_DOD = [od ; sum_ztot1];
    od = norm_Z_DOD; %8981x1
    do = [];
end
for m_dod= el1:el2
    if norm_Z_DOD(m_dod+1, :) - max(norm_Z_DOD) == 0
        elev_dod = m_dod;
    end
end
sudut_estdod(el) = elev_dod;
strvctr_dod = sv_linier (sudut_estdod(el));

% 3. Estimasi Delay
za = [];
for t = 1:K
    X_DEL = strvctr_doa'*
    X_ekpekstasi2(:, :, t)*conj(strvctr_dod);
    Z_DEL = [za X_DEL];
    za = Z_DEL; % 1 x K
end
del_ins = K;
for jj = 0:1:(del_ins-1)
    h = dfilt.delay (jj);
    delay = filter (h, bpsk(el, :));
    z_del_1 = delay;
    z_dell(jj+1, :) = z_del_1; % del_ins x K
end
dd = [];
for k = 1:del_ins
    X_delay = Z_DEL .* conj (z_dell(k, :));
    abs_delay = abs(sum(X_delay, 2).^2);
    Z_t_delay = [dd; abs_delay];
    dd = Z_t_delay;
end
for del = 1 : del_ins
    if Z_t_delay(del, :) - max(Z_t_delay) == 0
        DELAY = del-1;
    end
end
time_delay(el) = DELAY; % delay estimation

% terdelay time_delay1
h = dfilt.delay(time_delay(el));
states = h.states;
delay = filter (h, bpsk(el, :));
Xn_tau_new = delay;

```

```

pp=[];
for m = 1:N
    X_tau = X_ekspektasi(m,:).*conj (Xn_tau_new);
    X_ntau = [pp ; X_tau];
    pp = X_ntau; % N x K
end
X_ntau2 = reshape (X_ntau,Nr,Nt,K);
AMP=[];
for t = 1:K
    X_amp = strvctr_doa'*
X_ntau2(:,t)*conj(strvctr_dod);
    z_AMP = [AMP X_amp];
    AMP = z_AMP;
end
amp_complex(el) = (1/(Nr*Nt*K))*sum (z_AMP); % get
amp_complex_estimated
% Menghitung Error
sudut2 = sudut_estdoa(el) - sudut_estdoa1(el);
sudut1 = sudut_estdod(el) - sudut_estdod1(el);
time = time_delay(el) - time_delay1(el);
amp = amp_complex(el) - amp_complex1(el);
end % while (.....)
konvergen(v)= el;
sudut_estdoa1(el) = sudut_estdoa(el);
sudut_estdod1(el) = sudut_estdod(el);
time_delay1(el) = time_delay(el);
amp_complex1(el) = amp_complex(el);
% cetak jika el = L
if el == L
    for n = 1:L
        fprintf ('%d : %d : %d : %d
\t',sudut_estdoa1(n),sudut_estdod1(n),
time_delay1(n),amp_complex1(n));
    end
    fprintf ('\n')
end
end%v
% cetak hasil jika el ~= L
if el ~= L
    for n = 1:L
        fprintf ('%d : %d : %d : %d
\t',sudut_estdoa1(n),sudut_estdod1(n),
time_delay1(n),amp_complex1(n));
    end
    fprintf ('\n')
end
% Mencetak hasil jika nilai el ~= L
for n = 1:L
    fprintf
('-----\n',n)
;
    fprintf ('Sinyal ke %d.\n', n );
    fprintf
('-----\n',n)
;
    fprintf ('DOA = %d \n', gelev2(n));

```



```

fprintf ('DOA_estimasi = %d \n', sudut_estdoal(n))
fprintf ('DOD = %d \n', gelev1(n));
fprintf ('DOD_estimasi = %d \n', sudut_estdod1(n))
fprintf ('Delay = %d \n', gdelay(n))
fprintf ('Delay_estimasi = %d \n', time_delay1(n))
fprintf ('Amp = %d \n', abs(alfa(n)));
fprintf ('Amp_estimasi = %d \n', abs(amp_complex1(n)));

```

```
end
```

--- Listing Pembentukan Steering Vector ----

```

clc;clear all;close all
c = 3e8 ; L = 2; M = 4;
f = 2.4e9;
lamda = c/f;
beta = 2*pi/lamda;
r = 0.5*lamda;
P = [0 0 0; r 0 0; r r 0; 0 r 0 ];

Elmat = 0:1:180; % sudut elevasi array planar 90 derajat
Azimuth = 0:1:360; % azimuth array planar 360 derajat
mx = [];
for p = 1:length(Elmat)
    for q = 1:length(Azimuth)
        steering = sv_planar(Elmat(p),Azimuth(q));
        steer_matrix = [mx steering];
        mx = steer_matrix;
    end
end
str_planar = mx;

```

--- Listing Program Fungsi sv_linier dan sv_planar ----

```

function steering_vector = sv_linier (teta)
fi = 0;
c= 3e8 ; M = 2;
f= 2.475e9;
lamda=c/f;
beta=2*pi/lamda;
r=0.5*lamda;
TE_linier = [0 0 0; 0 0 r]; % untuk sv_linier
TE_planar = [0 0 0; r 0 0; r r 0; 0 r 0]; % untuk sv_planar

for jml = 1:length(teta)
    x(jml) = cos(fi*(pi/180))*sin(teta(jml)*(pi/180));
    y(jml) = sin(fi*(pi/180))*sin(teta(jml)*(pi/180));
    z(jml) = cos(teta(jml)*(pi/180));
    A = [x(jml) y(jml) z(jml)];

    for bar = 1:1:M
        B = TE_linier(bar,:);
        B = B.';
        C = A*B;
        steering_vector(bar,jml) = exp(j*beta*C);
    end
end

```

3. Data Simulasi merupakan nilai numerik hasil dari simulasi

✓ Antena Planar

- Analisa Perubahan SNR

Simulasi SAGE1

| snr | rasio_amp | doa_azim | doa_elev | dod_azim | dod_elev | delay |
|-----|-----------|----------|----------|----------|----------|--------|
| 0 | 0.0194 | 0.5495 | 0.3225 | 0.4427 | 0.3209 | 0.9803 |
| 5 | 0.0052 | 0.6066 | 0.1449 | 0.1049 | 0.1049 | 0.3661 |
| 10 | 0.0026 | 0.6099 | 0.1304 | 0.0837 | 0.0949 | 0 |
| 15 | 0.0006 | 0.1449 | 0.1449 | 0.0316 | 0.0548 | 0 |
| 20 | 0.0002 | 0.0548 | 0.1049 | 0.0316 | 0.0775 | 0 |
| 25 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30 | 0 | 0 | 0 | 0 | 0 | 0 |

Simulasi SAGE2

| snr | rasio_amp | doa_azim | doa_elev | dod_azim | dod_elev | delay |
|-----|-----------|----------|----------|----------|----------|--------|
| 0 | 0.9432 | 0.4393 | 0.253 | 0.3912 | 0.247 | 0.6317 |
| 5 | 0.6257 | 0.1581 | 0.1789 | 0.5206 | 0.1049 | 0.387 |
| 10 | 0.7731 | 0.1095 | 0.0707 | 0.0316 | 0.0447 | 0 |
| 15 | 0.5157 | 0.0316 | 0 | 0.0316 | 0 | 0 |
| 20 | 0.2313 | 0 | 0 | 0 | 0 | 0 |
| 25 | 0.1902 | 0 | 0 | 0 | 0 | 0 |
| 30 | 0.1529 | 0 | 0 | 0 | 0 | 0 |

- Analisa Perubahan Jumlah Path

Estimasi Delay

| L | SAGE2 | SAGE1 |
|---|--------|--------|
| 1 | 0 | 0 |
| 2 | 0.3373 | 0.3789 |
| 3 | 0.5302 | 0.6123 |
| 4 | 0.9172 | 0.9678 |
| 5 | 1.3781 | 1.3957 |
| 6 | 1.5158 | 1.6129 |

Estimasi Amplitudo

| L | SAGE1 amp1 | SAGE2 amp2 |
|---|---------------|---------------|
| 1 | 0.987 | 0.8888 |
| 2 | 0.7665 | 0.6665 |
| 3 | 0.6801 | 0.6301 |
| 4 | 0.6747 | 0.5467 |
| 5 | 0.674 | 0.5396 |
| 6 | 0.5816 | 0.4416 |

Estimasi Sudut (Elevasi)

| L | SAGE2 | | SAGE1 | |
|---|---------|---------|---------|---------|
| | elevdoa | elevdod | elevdoa | elevdod |
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0.2525 | 0.2236 | 0.2879 | 0.2632 |
| 3 | 0.2701 | 0.2674 | 0.3976 | 0.3764 |
| 4 | 0.4758 | 0.495 | 0.4253 | 0.4991 |
| 5 | 0.4491 | 0.7141 | 0.4982 | 0.7981 |
| 6 | 0.5734 | 0.8123 | 0.6762 | 0.899 |

Estimasi Sudut Azimuth

| L | SAGE2 | | SAGE2 | |
|---|---------|---------|---------|---------|
| | azimdoa | azimdod | azimdoa | azimdod |
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0.3303 | 0.2669 | 0.3678 | 0.3871 |
| 3 | 0.3737 | 0.5853 | 0.3998 | 0.6283 |
| 4 | 0.811 | 0.691 | 0.8972 | 0.7029 |
| 5 | 0.9301 | 1.0492 | 1.0235 | 1.0991 |
| 5 | 1.2932 | 1.0832 | 1.3098 | 1.1463 |

✓ Antena Linier

- Analisa Perubahan SNR

| snr | SAGE1 | | | | |
|-----|--------|--------|--------|--------|----------|
| | delay | doa1 | dod1 | rasio | rasio_db |
| 0 | 0.7242 | 0.148 | 0.1334 | 1.0122 | 0.052663 |
| 5 | 0.7114 | 0.1327 | 0 | 0.9996 | -0.00174 |
| 10 | 0.4645 | 0.1319 | 0 | 0.9999 | -0.00043 |
| 15 | 0.3637 | 0.0224 | 0 | 1.0001 | 0.000434 |
| 20 | 0.3635 | 0.02 | 0 | 1 | 0 |
| 25 | 0.1068 | 0.0141 | 0 | 1.0001 | 0.000434 |
| 30 | 0.1005 | 0 | 0 | 1 | 0 |

| snr | SAGE2 | | | | |
|-----|--------|--------|--------|--------|----------|
| | delay | doa2 | dod2 | rasio | rasio_db |
| 0 | 1.3809 | 0.3507 | 0.3231 | 1.0076 | 0.032882 |
| 5 | 0.824 | 0.2933 | 0.0316 | 1.0038 | 0.016472 |
| 10 | 0.7744 | 0.1902 | 0.0316 | 1.0008 | 0.003473 |
| 15 | 0.345 | 0.0707 | 0.0447 | 1.0003 | 0.001303 |
| 20 | 0.355 | 0.0632 | 0 | 1 | 0 |
| 25 | 0.3435 | 0.0568 | 0 | 1 | 0 |
| 30 | 0.3464 | 0 | 0 | 1 | 0 |

- Analisa Perubahan Jumlah Path

Simulasi SAGE1

| L | amp | elevasi_doa | elevasi_dod | delay |
|---|----------|-------------|-------------|--------|
| 1 | 9.89E-01 | 0.1732 | 0 | 0 |
| 2 | 0.8665 | 1.4491 | 0.1016 | 3.9984 |
| 3 | 0.6647 | 2.1909 | 0.1016 | 5.35 |
| 4 | 0.6567 | 2.6196 | 0.2271 | 8.5767 |
| 5 | 0.643 | 2.7372 | 0.2337 | 7.8773 |
| 6 | 0.5236 | 3.326 | 0.2645 | 9.6646 |

Simulasi SAGE2

| L | amp | elevasi_doa | elevasi_dod | delay |
|---|--------|-------------|-------------|--------|
| 1 | 0.9888 | 0.0071 | 0 | 0 |
| 2 | 0.8665 | 0.1056 | 0.1193 | 0.4023 |
| 3 | 0.6647 | 0.1881 | 0.1193 | 0.6266 |
| 4 | 0.6567 | 0.2241 | 0.2212 | 0.6464 |
| 5 | 0.643 | 0.2658 | 0.2201 | 0.7954 |
| 6 | 0.5236 | 0.291 | 0.2862 | 0.8367 |

Perubahan sudut Elevasi pada Antena Array Linier

| elevasi | SAGE1 | | SAGE2 | |
|---------|---------|--------|---------|---------|
| | doa | dod | doa | dod |
| 0 | 10 | 8.6023 | 8.3066 | 11.9583 |
| 1 | 10.198 | 8.4143 | 9.9499 | 8.4261 |
| 2 | 9.2195 | 0.7746 | 9.7468 | 0 |
| 3 | 12.3207 | 0 | 10.0598 | 0 |
| 4 | 10.02 | 0.4472 | 8.6833 | 0 |
| 5 | 8.7063 | 0 | 11.6017 | 0 |
| 6 | 6.261 | 0 | 10.2078 | 0 |
| 7 | 8.3427 | 0 | 5.9498 | 0 |
| 8 | 9.8995 | 0 | 7.6811 | 0.6325 |
| 9 | 1.5492 | 0 | 1.6125 | 0 |
| 10 | 1.8439 | 0 | 1.7889 | 0 |
| | | | | |
| 170 | 1.2649 | 0 | 0 | 0 |
| 171 | 2.3238 | 0 | 0 | 0 |
| 172 | 10.0499 | 0 | 0 | 0 |
| 173 | 8.0994 | 0 | 0 | 0 |
| 174 | 6.6182 | 0.4472 | 0 | 0 |
| 175 | 9.7263 | 0 | 6.3875 | 0 |
| 176 | 4.7329 | 5.9161 | 8.2098 | 0 |
| 177 | 9 | 0 | 8.5557 | 0 |
| 178 | 10.2372 | 0 | 8.438 | 0.4472 |
| 179 | 8.8091 | 6.0332 | 9.3059 | 8.4024 |
| 180 | 9.5917 | 0.7746 | 8.6487 | 6.0498 |

4. Preprocessing Data Pengukuran

• Listing Program Cek Autokorelasi

```

clc
clear all
close all
load data_m_8.txt
data = data_m_8(1:510,:).';
y = fft(data);
x = y.*conj(y);
ag = ifft(x);
ag2 = ag./max(ag);
figure,
plot(ag, 'LineWidth', 2)
axis([0 510 0 510 ])
grid on

```



```

title('Autokorelasi')
xlabel('sample')
ylabel('Autokorelasi Deretan PN yang dikirim')
figure,
stem(ag)
grid on
title('Autokorelasi Deretan PN yang Dikirim')
xlabel('sample')
ylabel('Autokorelasi')

```

- Listing Program Cek Crosscorelasi, Filtering Sausa, dan Pengecekan Konsistensi dan Tetangga

```

clc
clear all;
close all
load Y11_I.txt
load Y11_R.txt
load Y12_I.txt
load Y12_R.txt
load Y21_I.txt
load Y21_R.txt
load Y22_I.txt
load Y22_R.txt

H11 = j.*Y11_I + Y11_R ;
H21 = j.*Y21_I + Y21_R ;
H12 = j.*Y12_I + Y12_R ;
H22 = j.*Y22_I + Y22_R ;

H = [H11.'; H12.'; H21.' ;H22.'];
load data_m_8.txt
data_kirim = data_m_8(1:510,:);

data_kirim2 = data_kirim.';
K = length(data_kirim2);
datakirim_terdelay = [];
for jj = 0:1:K-1
    h = dfilt.delay (jj);
    delay = filter (h,data_kirim2);
    z_del_1 = delay;
    z_dell = [datakirim_terdelay ; z_del_1];
    datakirim_terdelay = z_dell;
end
% Crosscorelasi
cc = []; dd = [];
for k = 1: length(datakirim_terdelay)
    for p = 1:4
        X_Delay = H(p,:).*conj(datakirim_terdelay(k,:));
        X_Delay1 = [cc; X_Delay];
        cc = X_Delay1; % N x K
    end
end
cc = [];
X_Delay2 = abs(sum(X_Delay1,2)).^2;
X_tot = sum(X_Delay2,1);

```

```

X_tot2 = [dd X_tot];
dd = X_tot2;
end
X = X_tot2./max(X_tot2);
figure,
fs = 33e6; ts = 1/fs;
sb_x = ts*(0:1:length(X)-1);
plot(sb_x,X,'r')
title('Respon Impuls MIMO')
xlabel('excess delay')
ylabel('Daya ternormalisasi')
grid on

% cari nilai crocorelasi satu-satu
load satu1.txt
load satu2.txt
load dua1.txt
load dua2.txt
DATA = [satu1.'; dua1.'; satu2.'; dua2.'];
PP = []; op_geser = [];

% Filtering dengan sausa
for y = 1:4
    Pfa = 0.05; % 5%
    med_geser = median(DATA(y,:));
    stdvar = med_geser/sqrt(log(4));
    zeta = sqrt(-2*log(Pfa));
    threshold_geser = zeta * stdvar;
    simpan_threshold_geser = [threshold_geser PP];
    PP = simpan_threshold_geser ;
    % filtering dengan menggunakan sausa
    DATA2 = DATA(y,:);
    [a,b] = size(DATA2);
    for jj = 1:b;
        hasil_threshold_geser(:,jj) = DATA2(:,jj) .*
ge(DATA2(:,jj), threshold_geser);
    end

    hasil_threshold_ok = [ hasil_threshold_geser; op_geser];
    op_geser = hasil_threshold_ok;
end
DATA_SIAP_kons = op_geser;

figure,
H11_filt = DATA_SIAP_kons(1,:)./max(DATA_SIAP_kons(1,:));
plot(sb_x, H11_filt,'r')
grid on
title('H11 yang sudah terfilter')

figure,
H21_filt = DATA_SIAP_kons(2,:)./max(DATA_SIAP_kons(2,:));
plot(sb_x, H21_filt,'r')
grid on
title('H21 yang sudah terfilter')

```



```

figure,
H12_filt = DATA_SIAP_kons(3,:)./max(DATA_SIAP_kons(3,:));
plot(H12_filt,'m')
grid on
title('H12 yang sudah terfilter')

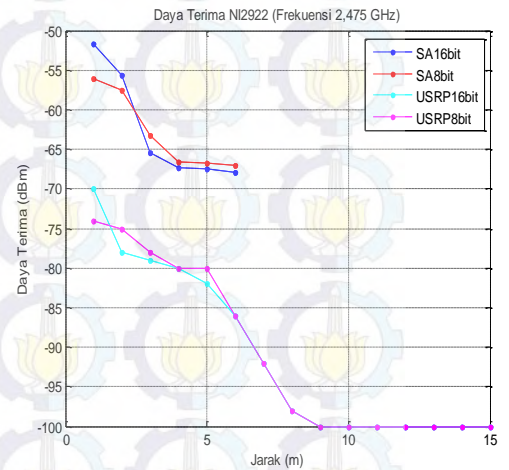
figure,
H22_filt = DATA_SIAP_kons(4,:)./max(DATA_SIAP_kons(4,:));
plot(sb_x, H22_filt,'r')
grid on
title('H22 yang sudah terfilter')
% Pengecekan Tetangga dan Konsistensi
DATA = DATA_SIAP_kons;
K = []; ol = [];
for i = 1:length(DATA)
    a = DATA(:,i);
    K = [];
    for k = 1:length(a)
        if a(k) ~= 0
            h = a(k);
        else
            h = 0;
        end
        h2 = [K; h];
        K = h2;
    end
    ki = find(h2~=0);
    jk = length(ki)/4;
    op = [ol ; jk];
    ol = op;
end
cek_konsistensi = op * 100;
time_delay1 = find(cek_konsistensi>=0.75);
L = length(time_delay1); fs = 33e6; ts = 1/fs;
sb_x = ts*(0:1:length(op)-1);
figure,
plot(sb_x,cek_konsistensi,'b')
grid on
title('Cek Konsistensi')
xlabel('Excess Delay')
ylabel('Prosentase Kemunculan(%)')

```

5. ile pendukung Pengukuran

✓ Sensitivitas USRP NI 2922 (Daya Terima)

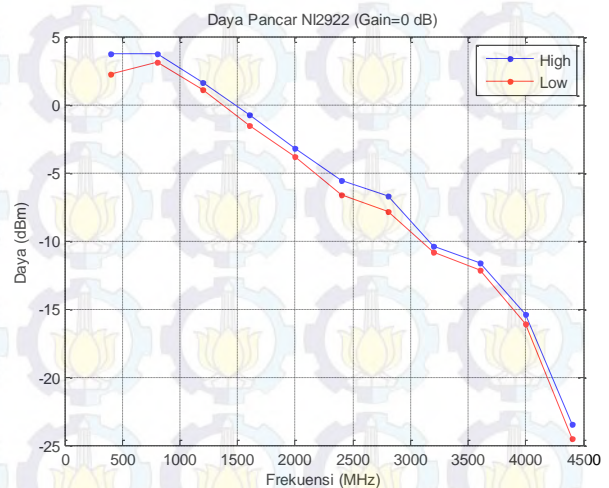
| Jarak(Km) | Pr Spektrum Analyzer (dBm) | | Pr USRP Rx | |
|-----------|----------------------------|--------|------------|-------|
| | 16 bit | 8 bit | 16 bit | 8 bit |
| 1 | -51,68 | -56 | -70 | -74 |
| 2 | -55,7 | -57,58 | -78 | -75 |
| 3 | -65,36 | -63,18 | -79 | -78 |
| 4 | -67,26 | -66,72 | -8- | -80 |
| 5 | -67,97 | -67,06 | -82 | -80 |
| 6 | | | -86 | -86 |
| 7 | | | -92 | -92 |
| 8 | | | -98 | -98 |
| 9 | | | -100 | -100 |
| 10 | | | -100 | -100 |
| 11 | | | -100 | -100 |
| 12 | | | -100 | -100 |
| 13 | | | -100 | -100 |
| 14 | | | -100 | -100 |
| 15 | | | -100 | -100 |



- IQ rate = 200k S/sec
- Frekuensi = 2,475 GHz
- Gain = 0

✓ Daya Transmit NI2922 (Frekuensi 400 MHz-4.4 GHz)

| Frekuensi (MHz) | Pout(dBm) | |
|-----------------|-----------|--------|
| | High | Low |
| 400 | 3,77 | 2,28 |
| 800 | 3,77 | 3,1 |
| 1200 | 1,64 | 1,08 |
| 1600 | -0,78 | -1,54 |
| 2000 | -3,19 | -3,8 |
| 2400 | -5,57 | -6,66 |
| 2800 | -6,67 | -7,86 |
| 3200 | -10,42 | -10,82 |
| 3600 | -11,64 | -12,16 |

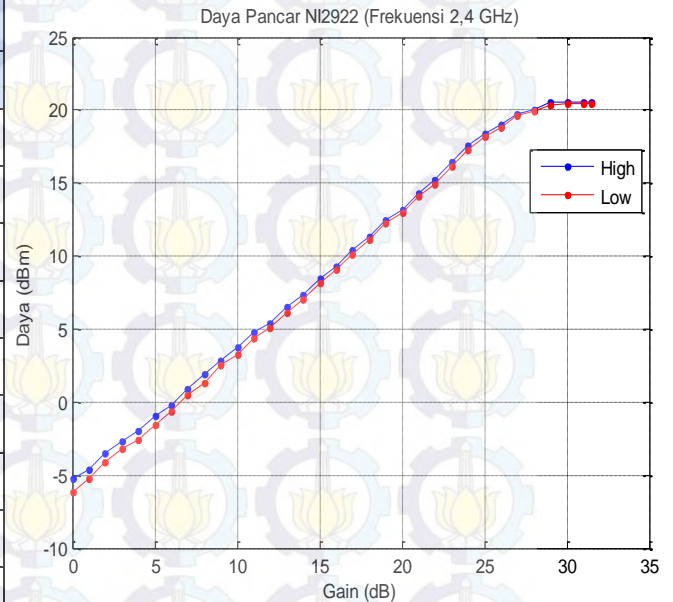


Gain = 0 Ref = 24 dBm
Att = 60 dB

| | | |
|------|--------|--------|
| 4000 | -15,42 | -16,1 |
| 4400 | -23,45 | -24,49 |

✓ Daya Transmit NI2922 (Gain 0 dB s/d 31.5 dB)

| Gain (dB) | P out (dBm) | | Gain (dB) | P out (dBm) | |
|-----------|-------------|-------|-----------|-------------|-------|
| | High | Low | | High | Low |
| 0 | -5,29 | -6,15 | 17 | 10,46 | 10,12 |
| 1 | -4,62 | -5,29 | 18 | 11,31 | 11,1 |
| 2 | -3,49 | -4,15 | 19 | 12,44 | 12,23 |
| 3 | -2,7 | -3,16 | 20 | 13,17 | 12,93 |
| 4 | -2 | -2,58 | 21 | 14,33 | 14,06 |
| 5 | -0,96 | -1,6 | 22 | 15,25 | 14,97 |
| 6 | -0,2 | -0,68 | 23 | 16,41 | 16,13 |
| 7 | 0,87 | 0,48 | 24 | 17,54 | 17,29 |
| 8 | 1,88 | 1,3 | 25 | 18,42 | 18,24 |
| 9 | 2,83 | 2,49 | 26 | 19,03 | 18,85 |
| 10 | 3,77 | 3,28 | 27 | 19,73 | 19,58 |
| 11 | 4,78 | 4,35 | 28 | 20,07 | 19,92 |
| 12 | 5,39 | 5,05 | 29 | 20,5 | 20,37 |
| 13 | 6,49 | 6,15 | 30 | 20,56 | 20,47 |
| 14 | 7,37 | 7,07 | 31 | 20,53 | 20,44 |
| 15 | 8,47 | 8,14 | 31,5 | 20,53 | 20,44 |
| 16 | 9,33 | 9,08 | | | |



- IQ rate = 1M S/sec
- Frekuensi = 2,4 GHz

✓ Link Budget (Free Space)

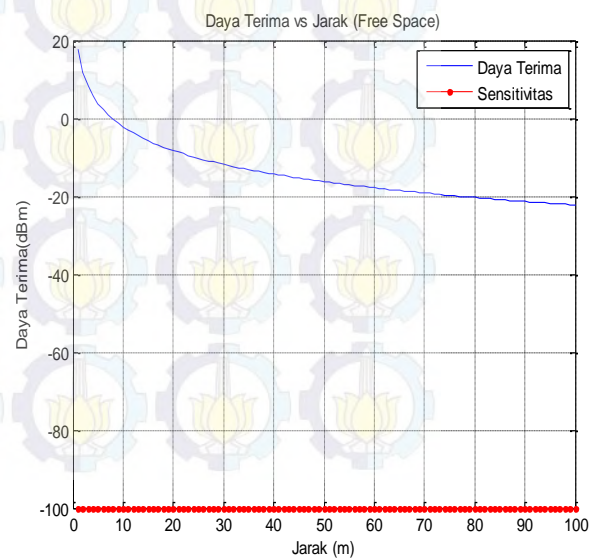
$$Pr = Pt + Gt + Gat + Gr - Lfsl$$

$$Lfsl = 32,4 + 20 * \log_{10}(d) + 20 * \log_{10}(f)$$

Dimana :

Pt : Daya Transmit

Gt : Gain USRP TX (31.5 dB)



Gat : Gain antenna pemancar (3dB)

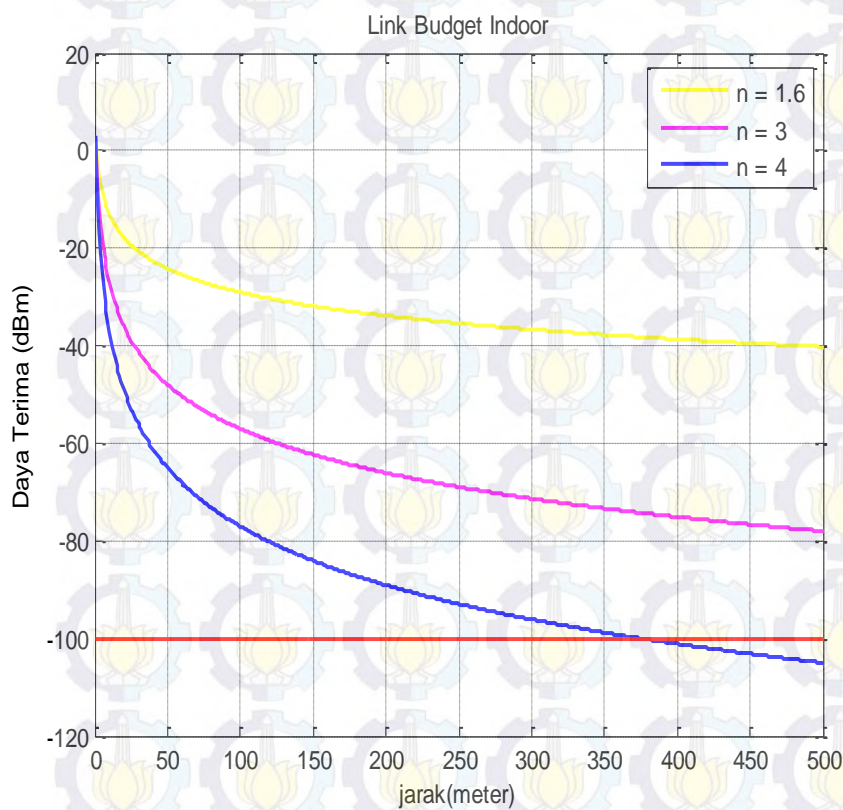
Gar : Gain antenna penerima (3dB)

Gr : Gain USRP RX (31.5 dB)

f : frekuensi (2474 MHz)

d : jarak (km)

✓ Link Budget (Indoor)



| Environment | n |
|---------------------------|------------|
| Free Space | 2 |
| In building line-of-Sight | 1.6 to 1.8 |
| Obstructed in building | 4 to 6 |

$$P_r(dB) = P_t(dB) + K(dB) - 10n \log_{10}\left(\frac{d}{d_0}\right)$$

$$K(dB) = 20 \log_{10}(\lambda / d_0 4\pi)$$

Dimana :

Pt : Daya Pancar (dB)

d : jarak (km)

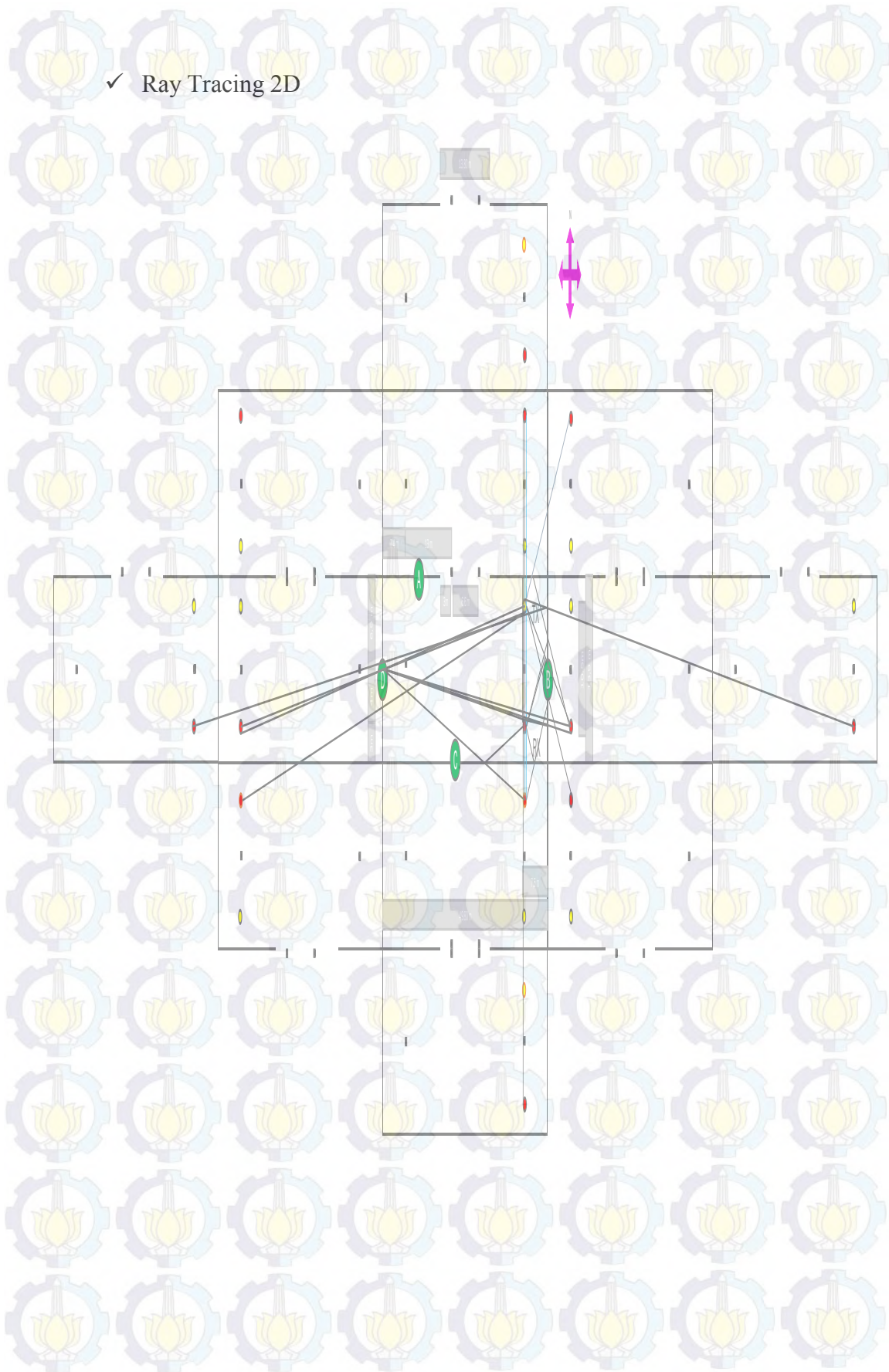
d₀ : jarak referensi (km)

n : Path Loss Exponen

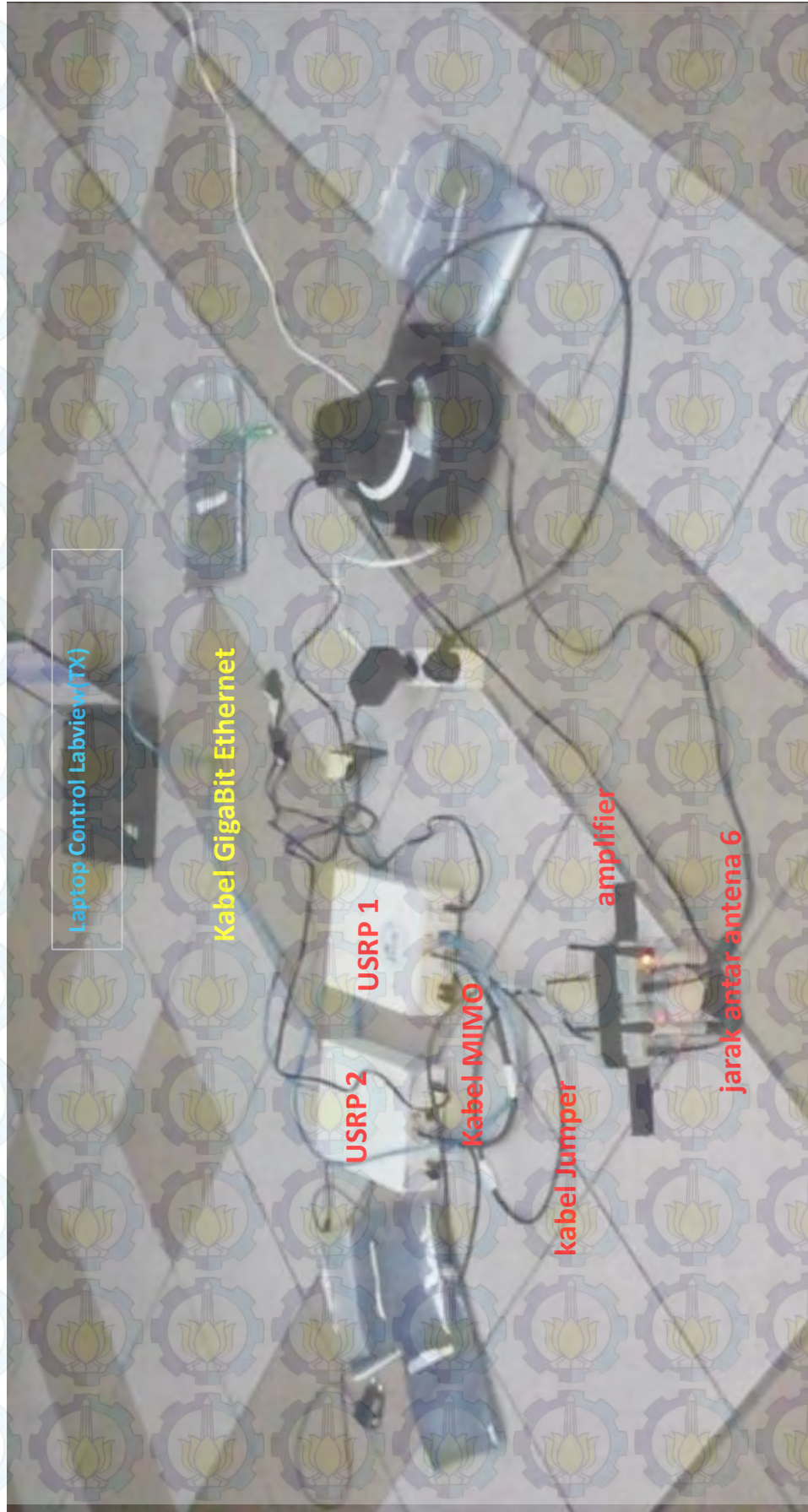
✓ Identifikasi Sinyal Ray Tracing 2 D (Hall, Gedung Baru PENS)

| Identifikasi Lintasan | Jarak (m) | Delay (ns) | Excess Delay (ns) | DOD (°) | DOA(°) |
|-----------------------|-----------|-------------|-------------------|---------|--------|
| LOS | 10.5 | 35 | 0 | 90 | 90 |
| 1 pantulan (A) | 15.3 | 51 | 16 | 90 | 90 |
| 1 pantulan (B) | 18.31 | 61.03333333 | 26.03333333 | 145 | 145 |
| 1 pantulan (C) | 12.5 | 41.66666667 | 6.666666667 | 90 | 90 |
| 2 pantulan (A,C) | 17.3 | 57.66666667 | 22.66666667 | 90 | 90 |
| 2 pantulan (C,A) | 38.3 | 127.66667 | 92.66666667 | 90 | 90 |
| 2 pantulan (A,B) | 21.43 | 71.43333333 | 36.43333333 | 134.43 | 134.43 |
| 2 pantulan (B,C) | 19.53 | 65.1 | 30.1 | 140.2 | 140.2 |
| 2 pantulan (D,C) | 85.52 | 285.0666667 | 250.0666667 | 8.4 | 8.4 |
| 2 pantulan (tiang,B) | 85.25 | 284.16667 | 249.1666667 | 7.07 | 172.93 |
| 2 pantulan (B,D) | 100.15 | 333.8333333 | | 173.98 | 173 |
| 2 pantulan (D,B) | 100.15 | 333.8333333 | 298.8333333 | 6.02 | 7.1 |

✓ Ray Tracing 2D



SETUP PENGUKURAN



RIWAYAT HIDUP



Penulis adalah lulusan dari D4 Jurusan Teknik Telekomunikasi PENS-ITS di tahun 2009, kemudian pada tahun 2013, penulis melanjutkan pendidikan S2 di Institut Teknologi Sepuluh Nopember, Bidang Studi Telekomunikasi Multimedia. Penulis mengikuti seminar dan ujian tesis dengan judul “**PENERAPAN SAGE (SPACE ALTERNATING GENERALIZED EXPECTATION-MAXIMIZATION) UNTUK ESTIMASI PARAMETER KANAL DUA ARAH (DOUBLE DIRECTIONAL) MENGGUNAKAN ANTENA ARRAY PLANAR**” sebagai salah satu syarat memperoleh gelar Magister Teknik (MT).

Nama : MUSAYYANAH

NRP : 2213203015

Email : musayyanah@gmail.com

Alamat : Kalimas Baru 2 No 06 Perak Utara Surabaya

Riwayat Pendidikan : 1. SDN Perak Utara 61 Surabaya (1997-2003)

2. SMP N 7 Surabaya (2003-2006)

3. SMA N 8 Surabaya (2006-2009)

4. D4 Teknik Telekomunikasi PENS (2009-2013)

5. S2 Telekomunikasi Multimedia ITS (2013-2015)